

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Методические указания

по дисциплине

«Интеллектуальные системы обработки информации и машинное
обучение»

для студентов

Специальности 10.05.03 Информационная безопасность
автоматизированных систем

Специализация «Анализ безопасности информационных систем»

Ставрополь
2026

ВВЕДЕНИЕ

Современные информационные технологии стремительно развиваются, и всё большее значение приобретают интеллектуальные системы, способные не только выполнять заданные алгоритмы, но и обучаться, адаптироваться и принимать решения в условиях неопределённости. Дисциплина «Интеллектуальные системы обработки информации и машинное обучение» направлена на формирование у студентов теоретических знаний и практических навыков, связанных с построением, применением и анализом интеллектуальных систем.

Интеллектуальные системы — это совокупность программных и аппаратных решений, которые моделируют когнитивные функции человека: распознавание образов, анализ данных, прогнозирование, принятие решений, обучение на основе опыта. Эти системы находят применение в самых разных отраслях: медицине, экономике, энергетике, образовании, промышленности, робототехнике, кибербезопасности и многих других.

Особое внимание в данной дисциплине уделяется машинному обучению — разделу искусственного интеллекта, в котором системы обучаются на основе данных и способны улучшать свои характеристики без явного программирования. Рассматриваются как базовые алгоритмы (линейные модели, деревья решений, кластеризация), так и более сложные методы, включая нейронные сети, глубокое обучение, методы обработки естественного языка и генеративные модели.

Кроме изучения теоретических основ, дисциплина акцентирует внимание на практических аспектах: работе с реальными данными, построении моделей, использовании современных аналитических платформ и библиотек (таких как Scikit-learn, TensorFlow, PyTorch, Deductor и др.), а также на решении прикладных задач с использованием интеллектуального анализа.

В рамках лабораторных работ студенты научатся:

- представлять и формализовать знания;
- использовать экспертные системы;
- применять методы классификации, регрессии, кластеризации и прогнозирования;
- работать с аналитическими платформами и визуализировать результаты;
- понимать ограничения алгоритмов и проводить их оценку по ключевым метрикам.

Изучение данной дисциплины формирует компетенции, необходимые для разработки интеллектуальных решений, анализа больших массивов данных и использования современных методов искусственного интеллекта в прикладной деятельности. Эти навыки особенно актуальны в условиях

цифровой трансформации и быстро меняющегося технологического ландшафта.

Лабораторная работа 1

Методы представления знаний в интеллектуальных системах: семантические сети и онтологии

Цель: Изучить основные методы представления знаний в интеллектуальных системах, познакомиться с понятийным аппаратом, связанным с семантическими сетями и онтологиями, освоить базовые принципы моделирования знаний в формализованной структуре.

Теоретическая часть

Представление знаний — один из ключевых аспектов построения интеллектуальных систем. Без чётко структурированной информации невозможна организация эффективного логического вывода, обучения и принятия решений. Методы представления знаний позволяют формализовать человеческие знания в виде, доступном для обработки компьютером.

Наиболее распространёнными формами представления знаний являются продукционные модели, фреймы, семантические сети и онтологии. Семантические сети представляют знания в виде ориентированных графов: вершины — понятия, рёбра — отношения между ними. Они удобны для визуализации иерархий, категоризаций и ассоциативных связей.

Онтологии, в свою очередь, позволяют строить более строгие и формализованные описания предметной области. В онтологиях задаются классы объектов, их свойства, отношения, ограничения и иерархии. Онтологии поддерживают автоматический вывод новых знаний, основанный на заложенных правилах и логике, и активно применяются в области семантического веба, медицины, биоинформатики, управления знаниями.

Форматы представления онтологий: OWL (Web Ontology Language), RDF, RDFS. Онтологические редакторы, такие как Protégé, позволяют визуально строить модели знаний и выполнять автоматическую проверку логической непротиворечивости.

Вопросы и задания

1. Объясните, зачем необходимо представление знаний в интеллектуальных системах.
2. Дайте определение семантической сети.

3. Перечислите основные элементы семантической сети и проиллюстрируйте их примерами.
4. В чём преимущество онтологии по сравнению с семантической сетью?
5. Какие реальные задачи можно решать с использованием онтологий?
6. В чём заключается роль отношений и ограничений в онтологиях?

Практические задания

1. Постройте семантическую сеть, моделирующую учебный процесс. Учтите такие сущности, как *курс, преподаватель, студент, задание, оценка* и взаимосвязи между ними (*читает курс, сдает задание, получает оценку* и т. д.).
2. В редакторе Protégé создайте онтологию предметной области «Интернет-магазин». Включите следующие классы: *Пользователь, Товар, Категория, Заказ*. Определите отношения, такие как *приобрел, содержит, принадлежит к категории*. Используйте свойства (атрибуты), например: *цена, дата, адрес доставки*.
3. Сравните семантическую сеть и онтологию по ключевым критериям: возможности масштабирования, формализация, применение логического вывода, гибкость расширения. Отрадите результаты в таблице.
4. Найдите в интернете пример открытой онтологии и кратко проанализируйте её структуру: какие классы, свойства и связи в ней реализованы.

Лабораторная работа 2

Основы программирования в «Пролог-Д»: логическое программирование и работа с базами знаний

Цель: Ознакомиться с принципами логического программирования, изучить структуру программы на языке Пролог-Д, освоить методы построения баз знаний и формулирования логических запросов.

Теоретическая часть

Логическое программирование — парадигма программирования, основанная на математической логике и автоматическом логическом выводе. В логических языках, таких как Пролог, разработчик формулирует базу знаний в виде фактов и правил, описывающих зависимости между объектами предметной области. Система автоматически определяет, выполняется ли заданный запрос, путём сопоставления с уже существующими знаниями.

Программа на Прологе — это совокупность предикатов. Факт представляет собой утверждение об объекте или связи между объектами. Правило задаёт

логическую зависимость между предикатами. Запрос позволяет извлекать информацию из базы знаний. Пролог выполняет унификацию и делает вывод на основе заданных правил.

«Пролог-Д» является отечественной реализацией языка и предоставляет расширенные возможности для построения интеллектуальных систем.

Пример:

отец(иван, алексей).
мать(мария, алексей).
родитель(X, Y) :- отец(X, Y).
родитель(X, Y) :- мать(X, Y).

При выполнении запроса родитель(X, алексей). система подставит в X всех, кто является родителем Алексея.

Вопросы и задания

1. В чём основное отличие логического программирования от процедурного?
2. Объясните понятия факт, правило и предикат на примере.
3. Как осуществляется логический вывод в Прологе?
4. Почему Пролог считается декларативным языком?
5. В каких задачах Пролог используется наиболее эффективно?

Практические задания

1. Установите среду программирования «Пролог-Д». Создайте базу знаний «Студенты и курсы». Опишите факты:

изучает(алексей, математика).
изучает(екатерина, информатика).
читает(иванов, математика).
читает(петрова, информатика).

2. Добавьте правило, определяющее предикат *учится_у*:

учится_у(Студент, Преподаватель) :- изучает(Студент, Курс),
читает(Преподаватель, Курс).

3. Выполните запрос:
 - а) Кто преподаёт Алексею?
 - б) Кто из студентов учится у Петровой?
4. Расширьте базу знаний, включив в неё курсы, объединённые по направлениям:

направление(математика, техническое).
направление(информатика, техническое).

Создайте правило студент_на_направлении (Студент, Направление) и проверьте, какие студенты обучаются по техническому направлению.

5. Напишите расширенное правило, определяющее коллег-преподавателей, читающих курсы одного направления.
6. В отчёте приведите код базы знаний, текст запросов и результаты выполнения. Оформите выводы по результатам лабораторной работы.

Вот расширенная лабораторная работа по теме "Разработка экспертной системы: идентификация проблемной области, концептуализация, формализация знаний и тестирование", оформленная в соответствии с принятым стилем:

Лабораторная работа 3

Разработка экспертной системы: идентификация проблемной области, концептуализация, формализация знаний и тестирование

Цель: Научиться определять проблемную область экспертной системы, выполнять концептуализацию и формализацию знаний, а также проводить тестирование созданной модели.

Теоретическая часть

Экспертная система (ЭС) — это программа, имитирующая процесс рассуждения и принятия решений эксперта в конкретной предметной области. Основу ЭС составляет база знаний, содержащая утверждения, правила и методы логического вывода, которые позволяют системе делать заключения и давать рекомендации.

Процесс разработки экспертной системы требует поэтапного подхода, включающего:

1. **Идентификацию проблемной области** — анализ сферы деятельности, в которой предполагается использовать ЭС, определение задач, которые должны решаться автоматически или с поддержкой со стороны ЭС. На этом этапе важно выделить границы системы, круг пользователей и цель функционирования.
2. **Концептуализацию знаний** — выделение ключевых понятий, объектов, сущностей и взаимосвязей между ними. Формируется общая логическая структура знаний в предметной области. Используются такие инструменты, как семантические сети, графы, онтологии, таблицы признаков.

3. **Формализацию знаний** — перевод выявленных знаний в форму, пригодную для машинной обработки. Это может быть продукционная форма (правила вида "ЕСЛИ–ТО"), фреймы (структурированные описания объектов), логические предикаты, деревья решений и др. На этом этапе создаётся база знаний, которая будет использоваться экспертной системой.
4. **Тестирование системы** — проверка корректности логического вывода, полноты и непротиворечивости базы знаний. Проводится на основе тестовых кейсов, заранее известных сценариев или при помощи диалогового взаимодействия с пользователем.

Разработка ЭС может вестись с использованием специализированных инструментов: CLIPS, Jess, Expert SNT, Gensym G2 и других.

Пример продукционного правила в условном синтаксисе:

ЕСЛИ температура > 38 И кашель = да
ТО диагноз = грипп.

Такая форма позволяет описывать медицинские, технические, юридические, финансовые и иные экспертные заключения.

Вопросы и задания

1. Что такое экспертная система и какие задачи она решает?
2. Как осуществляется выбор и ограничение проблемной области экспертной системы?
3. Что представляет собой концептуализация знаний?
4. Какие формы представления знаний используются в экспертных системах?
5. В чём отличие продукционного подхода от фреймового?
6. Как проводится тестирование экспертной системы? Приведите примеры.
7. Какие критерии можно использовать для оценки эффективности ЭС?

Практические задания

1. **Выбор проблемной области.** Выберите одну из предметных областей: медицина, техподдержка ИТ-систем, диагностика оборудования, профориентация, образовательные рекомендации. Определите задачу, которую должна решать экспертная система, опишите её с указанием предполагаемого круга пользователей.
2. **Концептуализация.** Выделите 6–8 ключевых понятий и объектов вашей предметной области. Нарисуйте семантическую сеть или таблицу взаимосвязей между ними.

Пример таблицы понятий для медицинской системы:

Симптом	Значение	Возможный диагноз
Температура > 38	Да	Грипп, ангина
Боль в горле	Да	Ангина, ОРВИ
Слабость	Да	Грипп, переутомление

3. **Формализация знаний.** Сформулируйте не менее 5 продукционных правил для своей системы. Укажите чёткие условия (ЕСЛИ) и заключения (ТО), оформите их в виде таблицы или отдельного списка.
4. **Реализация.** Используя среду CLIPS (или Jess, Expert SNT), реализуйте часть экспертной системы. Занесите базу знаний в виде фактов и правил. Настройте механизм логического вывода (forward chaining или backward chaining).
5. **Тестирование.** Проведите не менее 3-х тестовых сценариев. Проверьте, корректно ли система делает выводы при различных входных данных. Представьте результаты в виде таблицы:

Входные данные	Ожидаемый результат	Фактический результат	Совпадение
Температура = 39, Кашель = Да	Грипп	Грипп	Да
Температура = 36.5, Боль в горле = Да	ОРВИ	Ангина	Нет
Слабость = Да, Температура = 37.5	Переутомление	Переутомление	Да

Выводы. Подготовьте краткий отчёт: какие сложности возникли при формализации знаний, удалось ли системе воспроизвести экспертные решения, как можно улучшить структуру базы знаний.

Вот расширенная лабораторная работа по теме "**Аналитическая платформа Deductor. Хранилище данных**", оформленная в том же стиле:

Лабораторная работа 4

Аналитическая платформа Deductor. Хранилище данных

Цель:

Изучить архитектуру и возможности аналитической платформы Deductor, освоить понятие хранилища данных, понять принципы их проектирования и организации, а также научиться подключать и обрабатывать данные в Deductor Studio.

Теоретическая часть

Аналитическая платформа Deductor — это программное обеспечение российского производства, предназначенное для построения систем поддержки принятия решений на основе интеллектуального анализа данных (Data Mining) и бизнес-аналитики (BI). Deductor включает в себя ряд инструментов, которые позволяют проводить сбор, очистку, трансформацию, анализ и визуализацию данных.

Одним из ключевых компонентов платформы является **Deductor Studio** — визуальная среда, предназначенная для проектирования процессов обработки данных, моделирования и построения отчётов. С помощью Deductor можно реализовывать полноценные аналитические процессы без необходимости глубокого программирования.

Центральным объектом анализа в Deductor, как и в других BI-платформах, является **хранилище данных** (Data Warehouse). Хранилище представляет собой централизованную базу, куда поступают данные из различных источников: транзакционных систем, CRM, ERP, логов веб-приложений и др. Данные подвергаются трансформации и структурируются для дальнейшего анализа. В отличие от обычной БД, хранилища оптимизированы под чтение и анализ, а не под оперативную обработку транзакций.

Основные характеристики хранилищ данных:

- **Интеграция** — данные из разных источников объединяются в единую структуру.
- **Неподвижность (неизменяемость)** — данные после загрузки не изменяются, сохраняется история.
- **Временная зависимость** — данные фиксируются по времени, что позволяет анализировать изменения.
- **Предметная направленность** — данные организованы по аналитическим тематикам (продажи, клиенты и т. д.).

В Deductor предусмотрены средства визуального конструирования ETL-процессов (Extract, Transform, Load), которые позволяют собирать данные, очищать их, преобразовывать и записывать в хранилище.

Архитектура Deductor включает следующие компоненты:

- **Deductor Studio** — среда проектирования аналитических процессов.
- **Deductor Server** — серверная часть для хранения проектов, расчётов и автоматизации процессов.
- **Deductor Warehouse** — система хранения и обработки данных.

- **Библиотеки моделей и визуализации** — набор алгоритмов машинного обучения, средств построения диаграмм, графиков, OLAP-кубов и др.

Типичные шаги построения аналитического процесса в Deductor:

1. Подключение к источникам данных (файлы Excel, базы SQL, веб-сервисы).
2. Очистка и трансформация данных (преобразование типов, фильтрация, агрегация).
3. Загрузка данных в хранилище Deductor Warehouse.
4. Построение моделей анализа (кластеризация, регрессия, классификация).
5. Визуализация результатов в отчётах и дашбордах.

Вопросы и задания

1. Какие задачи решает аналитическая платформа Deductor?
2. Что такое хранилище данных и чем оно отличается от обычной БД?
3. Назовите и охарактеризуйте ключевые компоненты архитектуры Deductor.
4. В чём заключается этап ETL в построении аналитических систем?
5. Объясните различие между оперативной и аналитической обработкой данных.
6. Как осуществляется подключение внешних источников данных в Deductor?
7. Какие операции можно выполнять на этапе трансформации данных?
8. Что представляет собой Deductor Studio?
9. Какой формат используется для визуализации аналитических результатов?
10. Что такое OLAP и как его можно реализовать в Deductor?

Практические задания

1. **Изучение интерфейса Deductor Studio.**
Запустите Deductor Studio. Изучите основные панели: панель объектов, рабочее пространство, свойства узлов. Создайте новый проект. Добавьте несколько компонентов из библиотеки: источники данных, преобразования, визуализация.
2. **Подключение источника данных.**
Создайте соединение с внешним источником данных:
а) Excel-файл с таблицей продаж (предоставляется преподавателем);
б) база данных SQL Server (если доступно).
Проверьте подключение, просмотрите таблицы и выберите нужную для анализа.

3. Очистка и трансформация данных.

На основе импортированных данных выполните преобразования:

- фильтрация по дате;
- удаление пустых значений;
- преобразование форматов (даты, числа);
- создание нового вычисляемого столбца (например, прибыль = выручка – издержки).

4. Создание хранилища данных.

Импортируйте обработанные данные в структуру хранилища Deductor Warehouse. Создайте логическую схему: таблицы фактов и измерений, связи между ними.

Пример схемы хранилища:

Таблица фактов	Измерение 1	Измерение 2
Продажи	Продукты	Клиенты

Определите первичные ключи, типы связей, индексы.

5. Построение аналитического отчёта.

Создайте диаграмму или сводную таблицу, отображающую динамику продаж по месяцам и категориям товаров. Постройте график, отражающий зависимость прибыли от региона.

6. Автоматизация процессов.

Настройте расписание для ежедневной загрузки данных и их автоматической очистки. Проверьте работоспособность сценария.

7. Анализ результатов.

Сделайте вывод о целостности данных, удобстве представления, наличии выбросов и тенденций. Опишите, какие задачи бизнеса можно решать с помощью построенного отчёта.

Отчёт.

Подготовьте отчёт по проделанной работе, включив следующие разделы:

- описание источника данных;
- этапы обработки;
- схема хранилища;
- визуализации и диаграммы;
- выводы и рекомендации.

Лабораторная работа 5

Аналитическая платформа Deductor. Ассоциативные правила

Цель:

Изучить принципы построения ассоциативных правил, научиться применять алгоритмы ассоциативного анализа (например, Apriori) в среде аналитической платформы Deductor, освоить этапы подготовки данных и интерпретации полученных закономерностей.

Теоретическая часть

Ассоциативные правила представляют собой метод интеллектуального анализа данных, направленный на выявление закономерностей в виде правил вида «если А, то В», где А и В — совокупности элементов, часто встречающиеся вместе в базе транзакций. Этот метод широко применяется в маркетинге, розничной торговле, онлайн-аналитике и других сферах, где важно понимать поведение пользователей и предпочтения клиентов.

Одним из наиболее известных алгоритмов для извлечения ассоциативных правил является алгоритм Apriori. Его суть заключается в поэтапном расширении часто встречающихся наборов элементов и отсеке тех, которые не удовлетворяют заданным порогам поддержки (support) и доверия (confidence).

Поддержка — это доля записей, содержащих интересующий набор.

Доверие — это вероятность того, что элемент В встретится в записи при наличии элемента А.

Например, правило {Хлеб, Масло} → {Молоко} с поддержкой 20% и доверием 80% означает, что в 20% всех транзакций одновременно встречаются хлеб, масло и молоко, и в 80% случаев покупки хлеба и масла также покупается молоко.

Платформа Deductor предоставляет встроенный инструмент для работы с ассоциативными правилами, позволяющий пользователю визуально задать параметры анализа, выбрать поля, сформировать множество транзакций и получить набор ассоциаций. Одним из преимуществ Deductor является удобная настройка фильтров и возможность быстро изменять пороговые значения для генерации новых правил. Также Deductor обеспечивает визуализацию полученных закономерностей в табличной и графической форме.

Области применения ассоциативного анализа:

Рекомендательные системы (что купить вместе), выявление шаблонов мошенничества, анализ поведения пользователей на сайте, логистика и хранение товаров на складе, маркетинговые акции.

Процесс построения ассоциативных правил в Deductor:

Подключение источника транзакционных данных, трансформация данных в формат «объект — признаки», настройка параметров анализа (support, confidence, lift), запуск алгоритма и интерпретация результатов.

Вопросы и задания

1. Что такое ассоциативные правила и в каких случаях они применяются?
2. Опишите алгоритм Apriori.
3. Объясните понятия поддержка и доверие в контексте ассоциативных правил.
4. Какие существуют ограничения и потенциальные проблемы при использовании ассоциативного анализа?
5. Как выбрать поля для анализа в транзакционных данных?
6. Что такое метрика lift и как она интерпретируется?
7. Какова роль трансформации данных в процессе ассоциативного анализа?
8. Какие возможности предоставляет Deductor для визуализации правил?
9. В чём преимущество визуального анализа ассоциаций перед текстовым представлением?
10. Приведите примеры применения ассоциативных правил в реальных бизнес-задачах.

Практические задания

1. Откройте Deductor Studio и создайте новый проект. Импортируйте файл транзакций (например, данные о покупках клиентов, предоставленные преподавателем). Ознакомьтесь с полями и структурой таблицы.
2. Подготовьте данные к ассоциативному анализу. Сформируйте набор «объект – множество признаков», сгруппируйте данные по идентификатору транзакции. Убедитесь, что в каждой строке отображён один покупатель или транзакция с перечнем купленных товаров.
3. Настройте узел анализа ассоциаций. Задайте параметры:
 - минимальная поддержка: 0.1
 - минимальное доверие: 0.5
 - максимальная длина правила: 3
4. Запустите анализ и изучите полученные правила. Проанализируйте хотя бы 10 правил, выберите из них наиболее значимые по значениям поддержки, доверия и lift. Постарайтесь интерпретировать каждый из них: какие товары часто покупают вместе и как это может использоваться в бизнесе.
5. Постройте визуальное представление полученных правил. Используйте встроенные средства Deductor для создания графов связей или таблиц с фильтрацией. Настройте цветовые шкалы по значениям доверия или lift.
6. Измените пороговые значения (например, поддержку — до 0.05, доверие — до 0.4) и снова выполните анализ. Сравните полученные

результаты с предыдущими. Сделайте вывод о влиянии параметров на объём и качество правил.

7. Подготовьте отчёт по лабораторной работе, включив:
 - описание используемых данных;
 - этапы подготовки данных;
 - список полученных ассоциативных правил;
 - визуализации;
 - интерпретацию и возможное применение результатов анализа.
8. Дополнительное задание (по желанию): попробуйте использовать фильтрацию по конкретным товарам или группам товаров (например, продукты из категории «детское питание»), чтобы построить специализированные правила.

Вывод:

Выполнение данной лабораторной работы позволяет ознакомиться с базовыми методами ассоциативного анализа и освоить инструменты платформы Deductor для извлечения знаний из транзакционных данных. Полученные навыки применимы в сфере маркетинга, логистики, рекомендательных систем и могут использоваться при анализе клиентского поведения и разработке бизнес-стратегий.

Лабораторная работа 6

Аналитическая платформа Deductor. Прогнозирование

Цель:

Изучить методы прогнозирования на основе исторических данных с применением инструментов аналитической платформы Deductor. Освоить этапы подготовки данных, настройки моделей прогнозирования и интерпретации полученных результатов. Научиться применять алгоритмы машинного обучения и статистики для решения прикладных задач прогнозного анализа.

Теоретическая часть

Прогнозирование — это один из важнейших этапов анализа данных, направленный на предсказание будущих значений целевой переменной на основе имеющейся информации. В бизнесе и других сферах прогнозирование используется для планирования продаж, управления запасами, оценки спроса, прогнозирования прибыли и анализа поведения потребителей.

Платформа Deductor предоставляет пользователю гибкий инструментарий для решения задач прогнозирования с помощью различных моделей. Среди наиболее часто используемых методов в Deductor: линейная регрессия, деревья решений, ансамблевые методы, нейронные сети, а также алгоритмы экспоненциального сглаживания и авто-регрессионные модели.

Процесс построения прогностической модели включает несколько ключевых этапов:

1. Подготовка данных: очистка, нормализация, выбор признаков и временных рамок.
2. Разделение на обучающую и тестовую выборки.
3. Настройка параметров модели и её обучение на исторических данных.
4. Проверка точности модели (метрики: MAE, RMSE, R^2 и др.).
5. Построение прогноза на будущие периоды.
6. Интерпретация и визуализация результатов, принятие решений.

В Deductor реализована возможность работы как с временными рядами, так и с обобщёнными наборами данных. Интуитивно понятный интерфейс позволяет настраивать узлы анализа и визуализации, обеспечивая быстрый переход от данных к прогнозу.

Типичные задачи прогнозирования:

Прогноз продаж товаров в зависимости от сезонности, анализ посещаемости сайта, предсказание объёмов производства, оценка изменения цен, предсказание отказов оборудования.

Вопросы и задания

1. Что такое прогнозирование и как оно применяется в интеллектуальном анализе данных?
2. Какие методы прогнозирования поддерживаются в платформе Deductor?
3. В чём заключается разница между линейной регрессией и методами машинного обучения для прогнозирования?
4. Какие требования предъявляются к данным при построении модели прогнозирования?
5. Что такое переобучение и как его избежать?
6. Какие метрики качества моделей прогнозирования вы знаете?
7. Как интерпретировать коэффициент детерминации (R^2)?
8. В чём особенности работы с временными рядами?
9. Как в Deductor осуществляется визуализация прогноза?
10. Какие ограничения могут возникнуть при применении моделей прогнозирования на практике?

Практические задания

1. Создайте новый проект в Deductor Studio. Импортируйте датасет с историческими данными (например, месячные продажи продукции по регионам). Ознакомьтесь с его структурой и очистите от пропусков или некорректных значений.

2. Проведите предварительный анализ данных. Постройте графики распределения, изучите поведение целевой переменной во времени, определите наличие сезонности и тренда.
3. Разделите данные на обучающую и тестовую выборки (например, 80% на обучение и 20% на тест). Настройте модель линейной регрессии для прогнозирования на основе одного или нескольких признаков (время, регион, категория товара и т. д.).
4. Постройте прогноз на выбранный период. Постройте графики сравнения фактических и прогнозных значений. Вычислите точность модели по метрикам MAE (средняя абсолютная ошибка), RMSE (среднеквадратичная ошибка), R^2 (коэффициент детерминации).
5. Добавьте в проект альтернативную модель, например, дерево решений или градиентный бустинг. Повторите процесс обучения и сравните результаты с предыдущей моделью. Проанализируйте, какая модель показывает лучшую точность и устойчивость.
6. Настройте визуализацию результата прогнозирования в виде линейных графиков и таблиц. При необходимости настройте фильтры по группам товаров или регионам. Проведите аналитический обзор.
7. Измените параметры модели (например, количество признаков, глубину дерева, параметры регуляризации) и проанализируйте влияние этих изменений на итоговую точность прогноза.
8. Подготовьте письменный отчёт по работе, включив:
 - описание использованных данных;
 - этапы подготовки;
 - параметры моделей;
 - визуализации;
 - анализ результатов;
 - выводы о применимости метода в реальных задачах.
9. Дополнительное задание: примените алгоритм скользящего окна для прогноза временного ряда. Сравните этот подход с обучением на всем датасете сразу.

Вывод:

В рамках лабораторной работы были изучены основные принципы построения моделей прогнозирования в среде Deductor. Студенты получили практический опыт работы с историческими данными, научились выбирать подходящую модель, оценивать её точность и формировать обоснованные прогнозы. Навыки, полученные при выполнении данной работы, могут быть полезны при решении прикладных задач в бизнесе, экономике, маркетинге, логистике и других областях, требующих анализа динамики и предсказания будущих значений.