Министерство науки и высшего образования РФ.

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Лабораторная работа №1

по дисциплине

«Искусственный интеллект и машинное обучение»

«Первичный анализ данных»

|  |
| --- |
| Выполнил:  Окунев Николай Александрович  студент 2 курса группы КМБ-с-о-23-1 специальности 10.05.01 «Компьютерная безопасность»  очной формы обучения  (подпись)  Проверил:  Свидницкий Алексей Дмитриевич ассистент ДЦРСиЭ  (подпись) |

Ставрополь, 2025 г

Лабораторная работа 1.

Первичный анализ данных

**Цели и задачи работы**: изучение программных средств для

организации рабочего места специалиста по анализу данных и машинному обучению.

Основные задачи:

* получение программного доступа к данным, содержащимся в
* источниках различного типа;
* выполнение предварительного анализа данных и получение обобщенных характеристик наборов данных;
* исследование простых методов визуализации данных;
* изучение основных библиотек Python для работы с данными.

Ход выполнения работы:

Данный датасет содержит записи, фиксирующие действия игрока в игре OSU!. В каждой строке представлена следующая информация:

**Положение курсора:** Два числовых значения типа float, которые указывают координаты курсора по осям x и y. Это позволяет отслеживать перемещения курсора по экрану.

**Время с последнего действия:** Числовое значение, представляющее интервал времени в миллисекундах (ms) с момента предыдущего действия. Эта метрика полезна для анализа динамики и скорости реакций игрока.

**Название карты:** Текстовое поле, содержащее имя карты (уровня), с которой были получены данные. Это позволяет сегментировать и анализировать поведение игрока в зависимости от конкретных игровых сценариев.

Таким образом, датасет можно использовать для анализа игровых паттернов, изучения скорости и точности движений курсора, а также для сравнения результатов между различными картами.

Внешний вид данных:

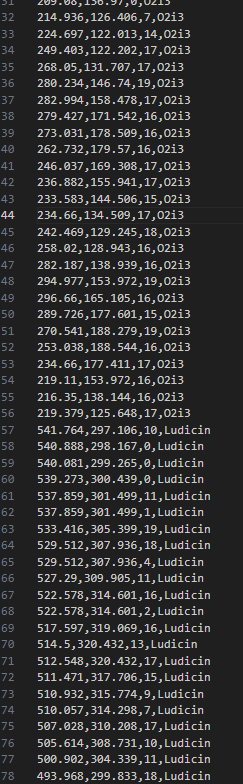


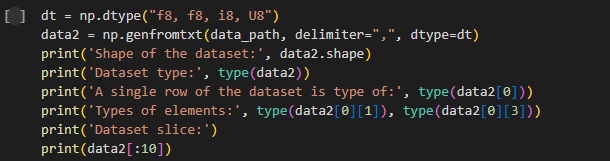
Рисунок 1 - внешний вид датасета

Для анализа набора данных использовался Jupyter Notebook.

Для работы с данными использовались библиотеки numpy, matplotlib, matplotlib.pyplot.

Вместо использования 4-x мерного пространства для работы с элементами, строились проекции для пар x – y, y – lat, x – lat.

Построение графиков при помощи Matplotlib



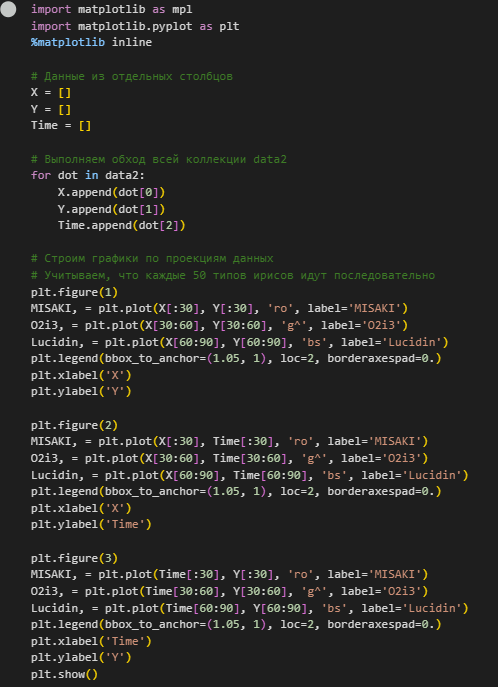
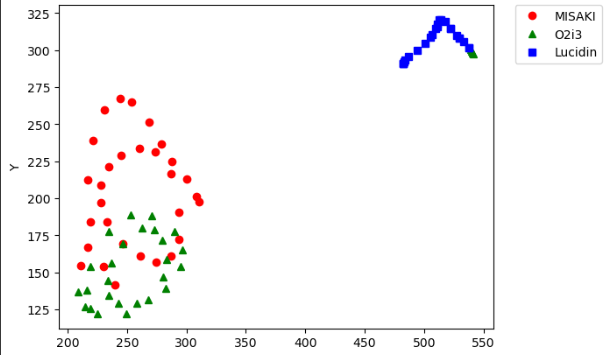


Рисунок 2 - построение графиков

Графики:



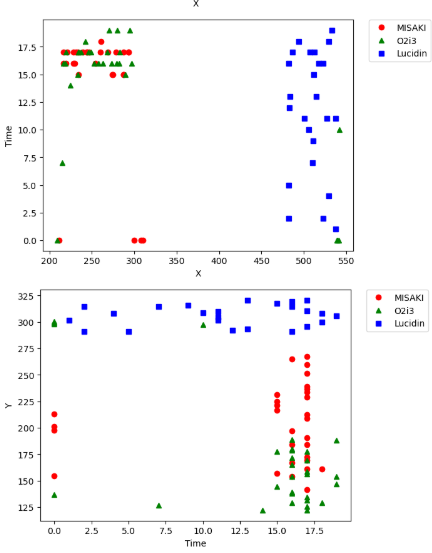


Рисунок 3 - Графики

Контрольные вопросы:

1)Какие инструментальные средства используются для организации рабочего места специалиста Data Science?

Языки программирования:

* Python: Самый популярный язык для Data Science. У него много библиотек для анализа данных, машинного обучения и визуализации.
* R: Язык для статистических вычислений и графики. Тоже очень популярен в Data Science.
* SQL: Для работы с базами данных и извлечения данных.
* Scala: Для работы с большими данными и Spark.
* IDE (интегрированные среды разработки):
* Jupyter Notebook: Очень удобная среда для интерактивной работы с кодом, визуализацией и документацией.
* Visual Studio Code (VS Code): Мощный редактор кода с поддержкой Python, R и других языков. У него много расширений для Data Science.
* PyCharm: IDE от JetBrains специально для Python.
* RStudio: IDE для R.

Библиотеки и пакеты:

* NumPy: Для работы с массивами и матрицами.
* Pandas: Для анализа и манипуляции данными.
* Scikit-learn: Для машинного обучения.
* Matplotlib и Seaborn: Для визуализации данных.
* TensorFlow и PyTorch: Для глубокого обучения.
* Statsmodels: Для статистического моделирования.

Инструменты для работы с большими данными:

* Apache Spark: Для обработки больших данных.
* Hadoop: Для хранения и обработки больших данных.
* Dask: Для параллельных вычислений в Python.

Системы контроля версий:

* Git: Для управления версиями кода и совместной работы.
* GitHub, GitLab, Bitbucket: Платформы для хостинга Git-репозиториев.

Инструменты для развертывания моделей:

* Docker: Для создания контейнеров с моделями.
* Kubernetes: Для управления контейнерами.
* Cloud platforms (AWS, Azure, GCP): Для развертывания моделей в облаке.

Инструменты для визуализации данных:

* Tableau: Для создания интерактивных дашбордов.
* Power BI: Еще один инструмент для дашбордов от Microsoft.

Другие инструменты:

* Anaconda: Платформа для управления пакетами и средами в Python и R.
* Slack, Microsoft Teams: Для общения и совместной работы.
* Trello, Jira: Для управления проектами и задачами.

Это только основные инструменты. Data Scientist может использовать и другие инструменты в зависимости от проекта и задач. Главное чтобы инструменты помогали ему анализировать данные, строить модели и получать полезные результаты.

2)Какие библиотеки Python используются для работы в области машинного обучения? Дайте краткую характеристику каждой библиотеке.

Scikit-learn:

Описание: Самая популярная библиотека для машинного обучения. В ней есть много алгоритмов для классификации, регрессии, кластеризации, уменьшения размерности и много чего еще. Еще в ней есть инструменты для оценки моделей, выбора параметров и предварительной обработки данных.

Особенности: Простая в использовании, хорошо документирована, много примеров.

TensorFlow:

Описание: Библиотека от Google для глубокого обучения. В основном используется для создания и обучения нейронных сетей.

Особенности: Очень мощная, гибкая, можно использовать для решения разных задач.

Keras:

Описание: Высокоуровневый API для работы с TensorFlow (и другими бэкендами). Упрощает создание и обучение нейронных сетей.

Особенности: Простая в использовании, позволяет быстро создавать сложные модели.

PyTorch:

Описание: Еще одна популярная библиотека для глубокого обучения. В основном используется в научных исследованиях.

Особенности: Гибкая, динамическая, хорошо подходит для экспериментов.

XGBoost:

Описание: Библиотека для градиентного бустинга. Очень хорошо работает на разных задачах.

Особенности: Быстрая, точная, много настроек.

LightGBM:

Описание: Еще одна библиотека для градиентного бустинга от Microsoft.

Особенности: Быстрая, использует меньше памяти, чем XGBoost.

CatBoost:

Описание: Библиотека для градиентного бустинга от Yandex. Хорошо работает с категориальными данными.

Особенности: Не требует предварительной обработки категориальных данных, устойчива к переобучению.

Statsmodels:

Описание: Библиотека для статистического моделирования. В ней есть много моделей для регрессии, анализа временных рядов и других задач.

Особенности: Подробные результаты, позволяет проводить статистический анализ.

NLTK:

Описание: Библиотека для обработки естественного языка. В ней есть инструменты для токенизации, стемминга, лемматизации, POS-теггинга и других задач.

Особенности: Много инструментов для работы с текстом.

SpaCy:

Описание: Еще одна библиотека для обработки естественного языка. Она более быстрая и эффективная, чем NLTK.

Особенности: Быстрая, хорошо подходит для больших объемов текста.

3)Почему при реализации систем машинного обучения широкое распространение получили библиотеки Python?

* Простота и читаемость: Python легко учить и читать. У него понятный синтаксис и много примеров. Это делает его доступным для людей с разным опытом.
* Большое количество библиотек: В Python есть много библиотек для машинного обучения, анализа данных, визуализации и других задач. Это позволяет Data Science решать разные задачи, не изобретая велосипед.
* Сообщество: У Python большое и активное сообщество. Это значит что всегда можно найти помощь, задать вопрос или поделиться своим опытом.
* Кроссплатформенность: Python работает на разных операционных системах (Windows, macOS, Linux). Это удобно для разработки и развертывания моделей.
* Гибкость: Python можно использовать для разных задач, от простых скриптов до сложных систем машинного обучения.
* Интеграция с другими языками: Python можно интегрировать с другими языками, например C++ и Java. Это позволяет использовать существующий код и библиотеки.
* Открытый исходный код: Python - это язык с открытым исходным кодом. Это значит что его можно использовать бесплатно и изменять под свои нужды.
* Поддержка со стороны крупных компаний: Google, Facebook и другие крупные компании используют Python для машинного обучения. Они разрабатывают и поддерживают библиотеки, что делает Python еще более популярным.

Вывод: В процессе выполнения работы мы изучили программные средства для организации рабочего места специалиста по анализу данных и машинному обучению, а так же рассмотрели пример первичного анализа данных.