ПРОЕКТ 1: Классификация цветов Ирисов

В данном проекте решается задача многоклассовой классификации видов цветов ирисов на основе их морфологических характеристик. Использован метод k-ближайших соседей (KNN) для построения модели предсказания. Достигнута точность классификации **95.56%** на тестовой выборке.

Введение

Задача: Многоклассовая классификация
Цель: Построение модели для автоматического определения вида ириса по характеристикам цветка
Актуальность: Задача представляет собой классический пример обучения с учителем и широко используется для тестирования алгоритмов машинного обучения

Данные

Набор данных: Iris dataset из пакета RDatasets
Объем данных: 150 наблюдений
Признаки:

- Длина чашелистика (SepalLength)
- Ширина чашелистика (SepalWidth)
- Длина лепестка (PetalLength)
- Ширина лепестка (PetalWidth)

Целевая переменная: Вид ириса (3 класса)

- setosa (50 образцов)
- versicolor (50 образцов)
- virginica (50 образцов)

Методы

Алгоритм

Использован метод **k-ближайших соседей (KNN)** с евклидовой метрикой расстояния

Предобработка данных

- Данные загружены в исходном виде
- Целевая переменная преобразована в категориальный тип
- Выполнено масштабирование признаков

Разделение данных

- **Обучающая выборка:** 70% (105 образцов)
- **Тестовая выборка:** 30% (45 образцов)
- **Стратификация:** Сохранено распределение классов

Результаты

Общая точность модели

Точность на тестовой выборке: 95.56%

Матрица ошибок

	Ground Truth		
Predicted	setosa	versicol…	virginica
setosa	15	0	0
versicol…	0	11	2
virginica	0	0	17

Точность по классам

```
Точность по классам:
setosa: 100.0%
versicolor: 100.0%
virginica: 89.47%
```

Эксперимент с разными значениями K: K = 1, Точность: 97.78% K = 3, Точность: 95.56%

Эксперимент с разными значениями K

```
K = 1, Точность: 97.78%
K = 3, Точность: 95.56%
K = 5, Точность: 91.11%
K = 7, Точность: 91.11%
K = 9, Точность: 93.33%
K = 11, Точность: 93.33%
```

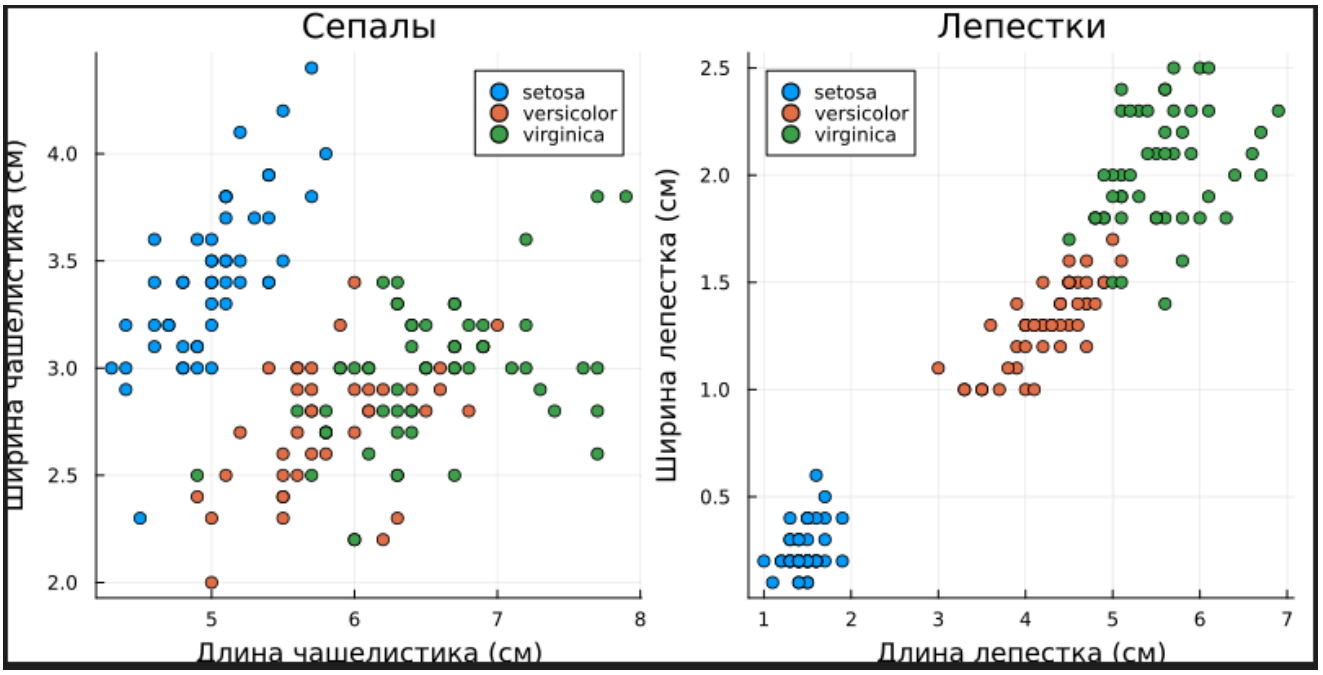
Анализ результатов

1. **Высокая точность:** Модель показывает точные результаты (97,78%)
2. **Лучший параметр:** K=1 демонстрирует наивысшую точность

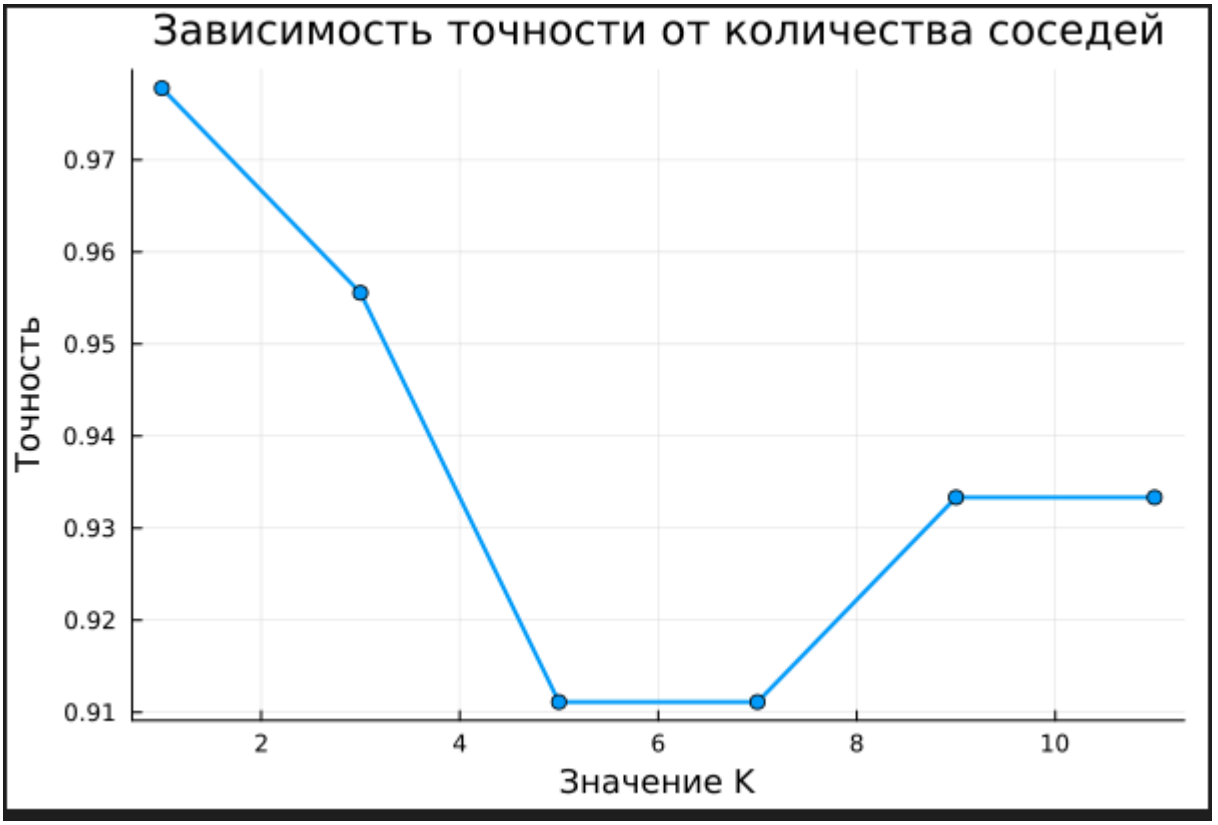
3. **Ошибки классификации:** Модель путает versicolor и virginica (1 ошибка в каждую сторону)
4. **Устойчивость:** Setosa идеально отделяется от других видов

Визуализация

Распределение данных



Зависимость точности от K



Заключение

Выводы

1. Метод KNN эффективен для решения задачи классификации ирисов
2. Оптимальное значение K = 1
3. Модель демонстрирует высокую точность (97,78%)
4. Наибольшую сложность представляет различение versicolor и virginica

Преимущества подхода

- Простота реализации
- Интерпретируемость результатов
- Высокая точность на данном наборе данных
- Не требует сложной предобработки

Ограничения

- Чувствительность к выбору параметра K
- Вычислительная сложность при больших объемах данных
- Зависимость от метрики расстояния

Перспективы развития

1. **Эксперименты с другими алгоритмами:**
   - Решающие деревья
   - Метод опорных векторов
   - Нейронные сети
2. **Улучшение модели:**
   - Подбор оптимальных параметров
   - Кросс-валидация
   - Ансамбли моделей
3. **Расширение функционала:**
   - Добавление новых признаков
   - Исследование feature importance
   - Разработка веб-интерфейса для предсказаний

Технические детали

Используемые технологии

- **Язык программирования:** Julia 1.9+
- **Библиотеки:** MLJ, MLJModels, NearestNeighborModels, Plots, DataFrames
- **Среда выполнения:** Jupyter Notebook

Время выполнения

- Обучение модели: < 1 секунды
- Предсказание: < 0.1 секунды

Приложение

Ключевые фрагменты кода

```
# Загрузка и подготовка данных
iris = dataset("datasets", "iris")
X = Matrix(iris[:, 1:4])
y = categorical(iris.Species)

# Создание и обучение модели
KNNModel = @load KNNClassifier pkg=NearestNeighborModels
knn = KNNModel(K=3)
knn_machine = machine(knn, X_train, y_train)
fit!(knn_machine)

# Оценка точности
accuracy = mean(mode.(y_pred) .== y_test)
```

Установка зависимостей

```
using Pkg
Pkg.add(["MLJ", "MLJModels", "NearestNeighborModels",
         "RDatasets", "DataFrames", "Plots", "CategoricalArrays"])
```

Дата выполнения: 06.09.2025
Выполнил: Киселев Георгий