

Министерство цифрового развития, связи и
массовых коммуникаций Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего
образования «Сибирский государственный университет телекоммуникаций и
информатики» (СибГУТИ)

Отчёт
по лабораторной работе №4
по дисциплине «**Прикладные задачи теории вероятностей**»

Выполнил:
студент гр. ИС-142
«__» декабря 2023 г.

/Григорьев Ю.В./

Проверил:
профессор кафедры В.С.,
«__» декабря 2023 г.

/Родионов А.С./

Оценка « _____ »

Новосибирск 2023

ВЫПОЛНЕНИЕ РАБОТЫ

Цель работы

Целью данной работы является изучение метода кластеризации k-средних и анализ влияния уровня разброса (стандартного отклонения) данных на качество кластеризации.

Генерация данных

Данные генерируются с использованием нормального распределения. Формула нормального распределения для генерации случайной величины X с математическим ожиданием μ и стандартным отклонением σ выглядит следующим образом:

$$X \sim N(\mu, \sigma^2)$$

Для каждого набора данных используется свое значение математического ожидания (1, 2, 3), а стандартное отклонение изменяется от 0.1 до 1.0.

Каждая выборка будет содержать 100 точек, распределенных по трем столбцам X_i, Y_i, Z_i (выборкам 1, 2, 3).

Кластеризация методом k-средних (K-means clustering)

Метод k-средних — это алгоритм кластеризации, который стремится разделить n наблюдений на k кластеров, в каждом из которых наблюдения находятся ближе к среднему (центроиду) кластера. Этот метод часто используется в машинном обучении для группировки данных без обучения человеком/программой-учителем.

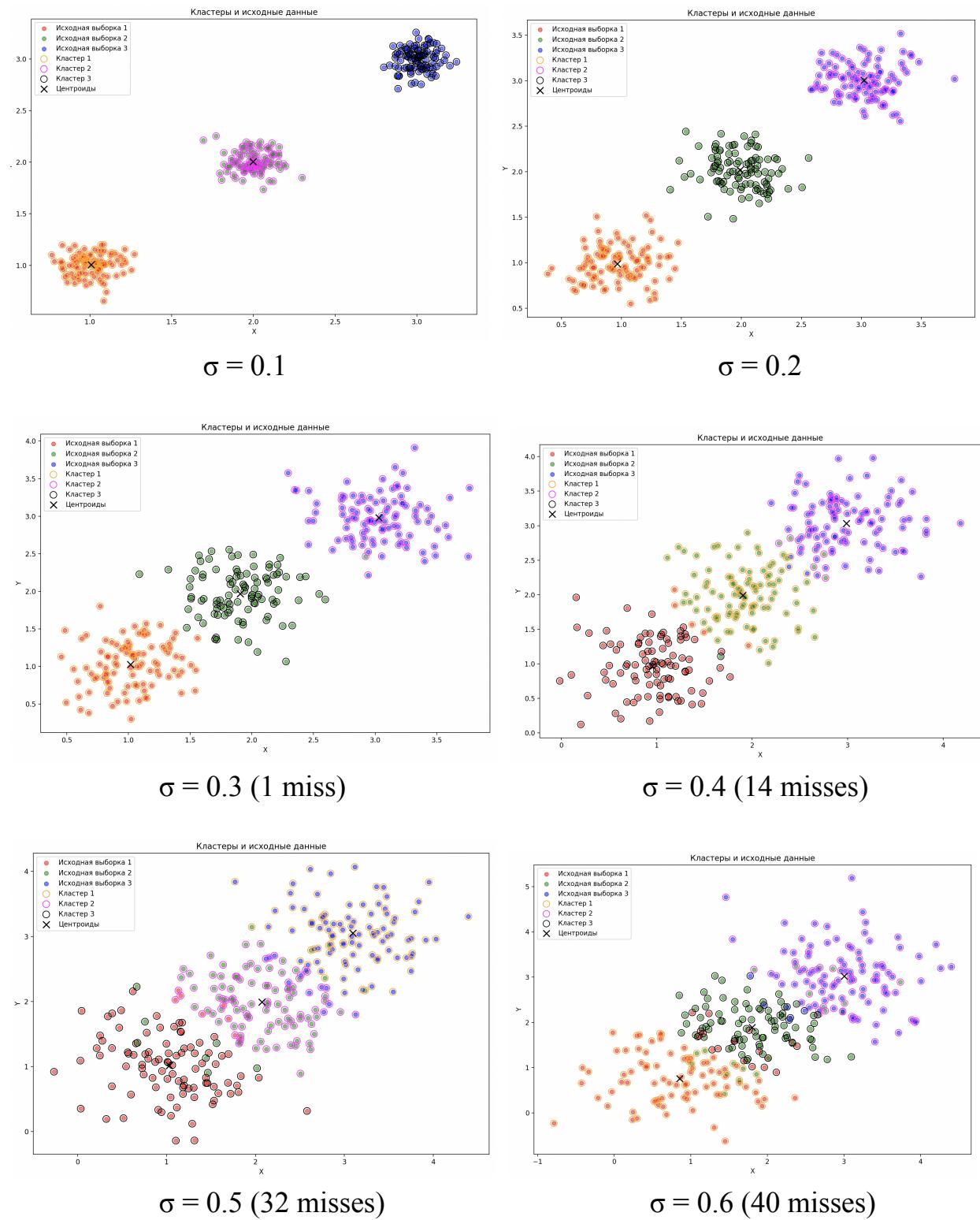
Основные шаги метода:

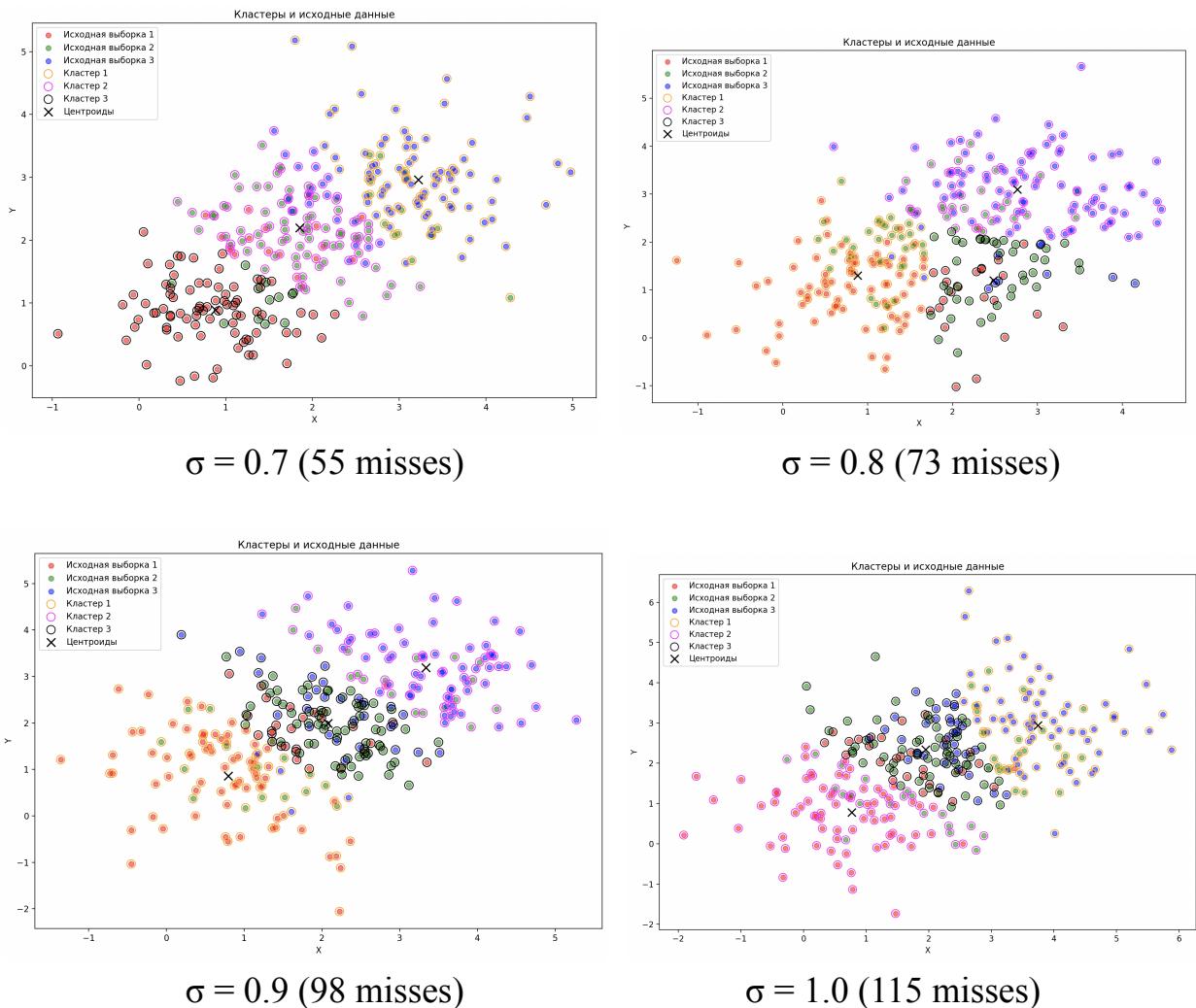
1. Выбор начальных центроидов (часто случайным образом).
2. Присвоение каждой точки данных к ближайшему центроиду.
3. Пересчет центроидов на основе принадлежности точек.
4. Повторение шагов 2-3 до сходимости (когда присвоение точек к кластерам больше не меняется).

Все эти шаги метода K-средних можно использовать в Python-библиотеке SciLearn, чем я и воспользуюсь. Кластеры отобразу цветной обводкой вокруг точек для наглядности.

Визуализация данных

Для визуализации данных используем библиотеку matplotlib. Данные из каждого набора будут визуализированы в двумерном пространстве с использованием различных цветов. Подпись “ N misses” говорит о том, что при кластеризации методом К-средних N точек выборок были отнесены не к своим изначальным кластерам.





Программа на Python для генерации данных, определения кластеров методом К-средних и визуализации полученных результатов

```

import matplotlib.pyplot as plt
import numpy as np
from sklearn.cluster import KMeans

# Генерация данных с матожиданиями 1, 2, 3 и отклонением sigma
means = [1, 2, 3]
sigma = 0.1
data_2d_sigma_05 = []

for mean in means:
    data_2d_sigma_05.append(np.random.normal(mean, sigma, (100, 2)))

data_2d_sigma_05 = np.concatenate(data_2d_sigma_05, axis=0)

# Применение метода k-средних
kmeans = KMeans(n_clusters=3)
kmeans.fit(data_2d_sigma_05)
labels = kmeans.predict(data_2d_sigma_05)

# Визуализация кластеров
plt.figure(figsize=(12, 8))

# Цвета для исходных данных
original_colors = ['red', 'green', 'blue']

```

```

# Отображение исходных данных с их цветами
for i in range(3):
    sample_data = data_2d_sigma_05[i*100:(i+1)*100]
    plt.scatter(sample_data[:, 0], sample_data[:, 1], alpha=0.5, label=f'Исходная выборка {i+1}', color=original_colors[i])

# Цвета для кластеров
cluster_colors = ['orange', 'magenta', 'black']

# Отображение кластеризованных данных с цветами кластеров
for i in range(3):
    cluster_data = data_2d_sigma_05[labels == i]
    plt.scatter(cluster_data[:, 0], cluster_data[:, 1], alpha=1.0, edgecolor=cluster_colors[i], facecolor='none', s=100, label=f'Кластер {i+1}')

# Отображение центроидов кластеров
centroids = kmeans.cluster_centers_
plt.scatter(centroids[:, 0], centroids[:, 1], color='black', marker='x', s=100, label='Центроиды')

plt.title('Кластеры и исходные данные')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()
plt.show()

# Определение точек, попавших не в свой кластер
misplaced_points = []
for i in range(3):
    original_sample = data_2d_sigma_05[i*100:(i+1)*100]
    misplaced = original_sample[labels[i*100:(i+1)*100] != i]
    misplaced_points.append(misplaced)

# Подсчёт количества ошибочно классифицированных точек в каждом кластере
misplaced_points_counts = [len(mp) for mp in misplaced_points]

```

Выходы

1. Чувствительность к стандартному отклонению

Один из самых заметных выводов из этой работы заключается в том, что точность кластеризации методом k-средних чувствительна к величине стандартного отклонения в данных. При малых значениях стандартного отклонения (например, 0.1) кластеры формируются более четко и точно. Однако с увеличением стандартного отклонения (например, до 0.5) наблюдается увеличение количества ошибочных классификаций. Это может быть связано с тем, что при большем разбросе данных увеличивается перекрытие между кластерами, что затрудняет их различие.

2. Влияние начального выбора центроидов

Метод k-средних начинается с произвольного выбора начальных центров кластеров, что может влиять на конечный результат. В некоторых случаях, особенно при высоком уровне разброса данных, начальный выбор центроидов может привести к существенно различным результатам кластеризации. Это подчеркивает необходимость многократного выполнения

алгоритма с разными начальными условиями для получения наиболее надежных результатов.

3. Применение в реальных сценариях

На основе проведенного анализа можно предположить, что метод k-средних будет наиболее эффективен в ситуациях, где данные хорошо разделены и имеют низкое стандартное отклонение. В сценариях с высоким уровнем шума или значительным перекрытием между классами может потребоваться применение более сложных методов кластеризации или предварительная обработка данных.

В целом, результаты показывают, что метод k-средних представляет собой мощный инструмент для кластеризации данных, но его эффективность может варьироваться в зависимости от конкретных характеристик анализируемого набора данных.