МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт Информационных Технологий

Кафедра МПО ЭВМ

Дисциплина «Математические методы решения задач искусственного интеллекта»

Лабораторная работа №4

«Анализ данных»

Исполнитель:

студент группы 1ПИб-02-3оп-22

Маркелов Сергей Александрович

Руководитель:

Юдина Ольга Вадимовна

2024 год

Вам предоставлен набор данных социологического опроса, проведенного кадровой службой предприятия. Работающие отвечали на вопросы о своих ценностях (применительно к работе на предприятии) и оценивали свою готовность остаться работать на предприятии, если оно сменит место дислокации (город на пригород)

Предприятию важно понять, сможет ли оно сохранить работоспособный коллектив в случае такого переезда, и какую программу поддержки следует разработать

Задание

Для поддержки аналитической работы постройте:

* Регрессию, которая содержит как количественные, так и качественные переменные
* Кластеризацию (на основе метода ближайших соседей)
* Классификацию (на основе метода ближайших соседей)

Точность и адекватность всех методов должны быть оценены

В отчете – пояснения к выбору признаков и методу (методам) выбора расстояний.

Для начала изучим данные в таблице. Она содержит:

Категориальные переменные:

* Пол (м/ж)
* Должность (рабочий, специалист, руководитель)
* Значимые ценности (высокая з/п, коллектив, содержание)
* Образование (среднее/высшее)
* Готовность остаться (да/нет/возможно)

Количественные переменные:

* Возраст
* Длительность работы
* Средняя з/п

1. Регрессия

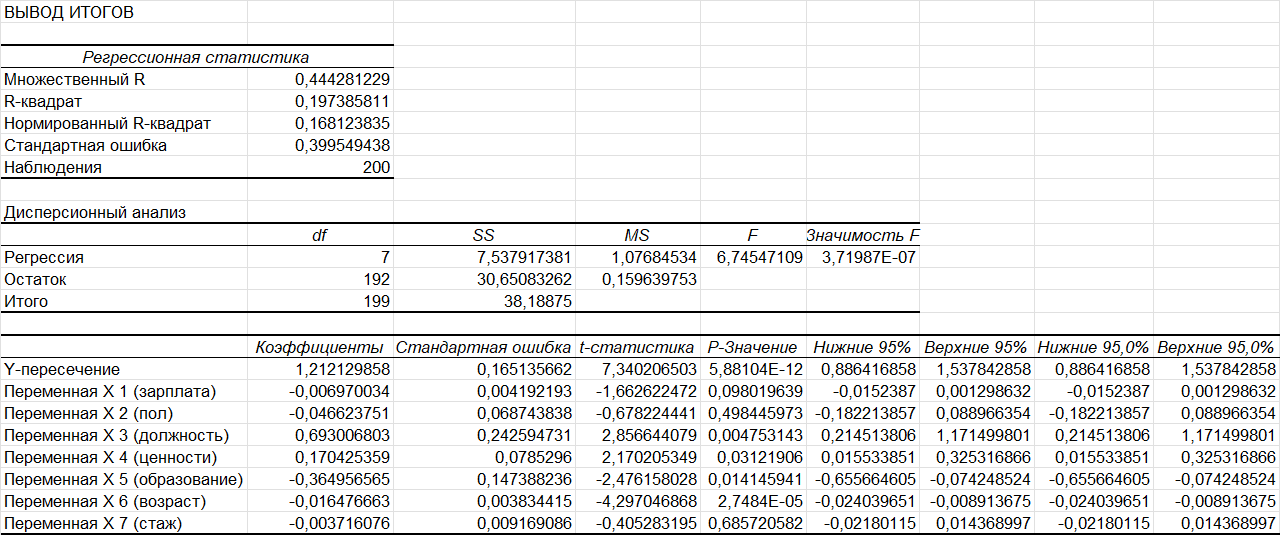
Средствами пакета анализа Excel построим регрессию, где:

Готовность остаться – зависимая переменная (1 – да, 0,5 – возможно, 0 – нет)

Все остальные переменные – независимые:

* Пол (1 – м, 0 – ж)
* Должность (0 – рабочий, 0,5 – специалист, 1 – руководитель)
* Значимые ценности (1 – высокая з/п, 0,5 – коллектив, 0 – содержание)
* Образование (0 – среднее, 1 – высшее)

Получаем следующий результат:



Проанализируем результаты.

Х1 – зарплата. Коэффициент -0,006970034 – отрицательный. Это значит, что при увеличении зарплаты снижает готовность сотрудника остаться. При этом значение коэффициента очень низкое, что говорит о незначительности влияния на готовность сотрудников остаться на предприятии.

Х2 – пол. Коэффициент -0,046623751 – отрицательный. Это значит, что пол имеет некоторое влияние на готовность остаться, но довольно слабое, т. к. коэффициент низкий. Следовательно, мужчины (1) имеют чуть меньшую вероятность остаться, чем женщины (0).

Х3 – должность. Коэффициент 0,693006803 – положительный. Это значение говорит о том, что должность оказывает влияние на готовность остаться, притом очень значительное, т. к. значение коэффициента высокое. Чем выше должность (чем ближе к 1), тем выше шанс, что сотрудник останется.

Х4 – ценности. Коэффициент 0,170425359 – положительный. Если сотрудник ориентирован на высокую зарплату (1), это увеличивает вероятность того, что он останется. Однако влияние умеренное, так как коэффициент не очень большой. Если приоритетом является коллектив или содержание, то это будет иметь меньший эффект на готовность остаться.

Х5 – образование. Коэффициент -0,364956565 – отрицательный. Это указывает на то, что люди с высшим образованием (1) с меньшей вероятностью будут готовы остаться, чем люди со средним образованием (0).

Х6 – возраст. Коэффициент -0,016476663 – отрицательный. Это говорит о том, что с возрастом вероятность остаться снижается. При этом коэффициент крайне низкий, следовательно, данный фактор почти не влияет на готовность сотрудников остаться на предприятии.

Х7 – стаж. Коэффициент -0,003716076 – отрицательный. Это также говорит о том, что с увеличением стажа вероятность того, что сотрудник останется на предприятии, уменьшается. Однако, коэффициент также низкий, поэтому данный фактор мало влияет на готовность сотрудников остаться на предприятии.

Итоги:

* Ключевое влияние на решение оказывает должность
* Также влияние оказывают образование и ценности
* Влияние зарплаты, пола, возраста и стажа мало или незначительно.

Оценим точность и адекватность метода.

R2 – коэффициент детерминации. Он показывает, какая доля изменчивости зависимой переменной объясняется моделью. Чем ближе R2 к 1, тем лучше модель объясняет данные.

В нашем случае R2 = 0,197385811. Следовательно, модель плохо описывает данные и нуждается в доработке.

Также проверим значимость переменных по p-значению. За уровень значимости примем α = 0,05. Это значит, что если p < 0,05, то переменная статистически значима

Х1 (зарплата) – 0,098019639 > 0,05 => статистически незначима, влияние не оказывает

Х2 (пол) – 0,498445973 > 0,05 => статистически незначима, влияние не оказывает

Х3 (должность) – 0,004753143 < 0,05 => статистически значима, оказывает влияние

Х4 (ценности) – 0,03121906 < 0,05 => статистически значима, оказывает влияние

Х5 (образование) – 0,014145941 < 0,05 => статистически значима, оказывает влияние

Х6 (возраст) – 2,7484E-05 < 0,05 => статистически значима, оказывает сильное влияние

Х7 (стаж) – 0,685720582 > 0,05 => статистически незначима, влияние не оказывает

1. Кластеризация

KNN (K-Nearest Neighbors) – это один из простых и популярных методов классификации и регрессии в машинном обучении. Основная идея метода заключается в том, чтобы для нового объекта (который нужно классифицировать или предсказать) найти его ближайших соседей в обучающей выборке и использовать их метки (для классификации) или значения (для регрессии) для получения прогноза.

Кластеризация – это задача, в которой модель пытается группировать объекты, которые похожи друг на друга, в кластеры, без заранее известных меток.

Для проведения кластеризации методом KNN воспользуемся средствами языка программирования Python. Напишем программу, разделяющую данные на кластеры.

import pandas as pd

from sklearn.preprocessing import StandardScaler

from sklearn.cluster import DBSCAN

from sklearn.metrics import silhouette\_score

# Загрузка данных

data = pd.read\_excel('Data.xlsx')

# Переменные

categorical\_columns = ['Пол', 'Должность', 'Значимые ценности', 'Образование']

numerical\_columns = ['Возраст', 'Длительность работы', 'Средняя з/п, мес., тыс. руб']

# Преобразуем все данные в числовой формат и нормализуем

X = data[categorical\_columns + numerical\_columns]

X\_normalized = StandardScaler().fit\_transform(X)  # Нормализация данных

# Применение DBSCAN

dbscan = DBSCAN(eps=0.5, min\_samples=5)

data['Cluster'] = dbscan.fit\_predict(X\_normalized)

# Анализ кластеров

sil\_score = silhouette\_score(X\_normalized, data['Cluster'])

print(f"Силуэтный коэффициент: {sil\_score}")

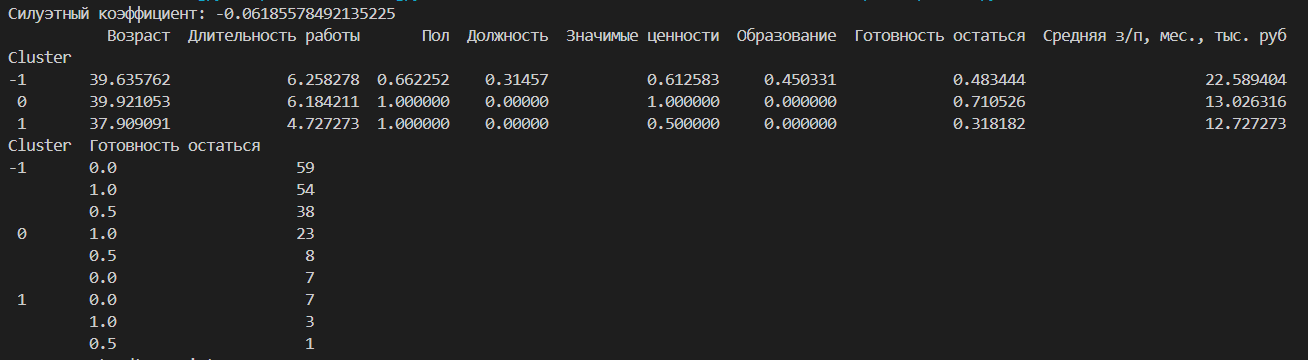
# Анализ средних значений по кластерам

print(data.groupby('Cluster').mean())  # Среднее по признакам для каждого кластера

# Смотрим, какие кластеры наиболее склонны остаться

print(data.groupby('Cluster')['Готовность остаться'].value\_counts())

Результат программы:



Всех сотрудников разбили на 3 кластера: -1, 0 и 1.

Кластер -1:

* Средний возраст – 39,64 лет
* Средний стаж – 6,25 лет
* Пол – 66,23% - мужчины
* Должность – 31,46% - руководители
* Значимые ценности – 61,26% - высокая з/п
* Образование – 45,03% - высшее
* Средняя з/п – 22589 руб.
* Готовность остаться – 48,34%
* 54 человека готовы остаться
* 59 человек не готовы остаться
* 38 человек не определились

В этом кластере наблюдается довольно сбалансированное распределение готовности остаться с преобладанием тех, кто все же готов остаться. Люди имеют средний возраст и стаж, зарплата выше по сравнению с другими кластерами.

Кластер 0:

* Средний возраст – 39,92 лет
* Средний стаж – 6,18 лет
* Пол – 100% – мужчины
* Должность – 0% – руководители
* Значимые ценности – 100% – высокая з/п
* Образование – 0% – высшее
* Средняя з/п – 13026 руб.
* Готовность остаться – 71,05%
* 23 человека готовы остаться
* 7 человек не готовы остаться
* 8 человек не определились

Этот кластер состоит в основном из мужчин со средним образованием, ориентированных на высокую зарплату. Эти сотрудники в основном готовы остаться, но их зарплата ниже, чем в кластере -1.

Кластер 1:

* Средний возраст – 37,91 лет
* Средний стаж – 4,73 лет
* Пол – 100% – мужчины
* Должность – 0% – руководители
* Значимые ценности – 50% – высокая з/п
* Образование – 0% – высшее
* Средняя з/п – 12727 руб.
* Готовность остаться – 31,82%
* 3 человека готовы остаться
* 7 человек не готовы остаться
* 1 человек не определился

В этом кластере меньше людей, готовых остаться, с меньшим средним возрастом и стажем. Среди них также преобладают мужчины со средним образованием, но они более ориентированы на зарплату.

Выводы:

* Кластер -1 – это группа, где присутствует большинство сотрудников, которые готовы остаться на предприятии, но есть и те, кто не готов или сомневается.
* Кластер 0 – группа, где сотрудники в основном готовы остаться, ориентированы на коллектив и содержание и имеют среднее образование. Однако их зарплата ниже, чем у других кластеров.
* Кластер 1 – это группа, где большинство сотрудников не готовы остаться, что может указывать на недостаток мотивации или неудовлетворенность работой.

Оценим точность и адекватность модели.

Силуэтный коэффициент – это мера качества кластеризации. Его значение варьируется от -1 до 1. Чем ближе к 1, тем лучше кластеризация.

В нашем случае силуэтный коэффициент: -0.06185578492135225. Это говорит о том, что кластеризация получилась слабой и неопределенной. Выбранный метод кластеризации не идеально разделяет данные на группы.

1. Классификация

Классификация – это задача, в которой у нас есть набор данных с известными метками классов, и мы обучаем модель, чтобы она могла предсказать эти метки для новых, неизвестных данных.

Для проведения классификации методом KNN также воспользуемся средствами языка программирования Python. Напишем программу, производящую классификацию данных.

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.neighbors import KNeighborsClassifier

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

# Загрузка данных

data = pd.read\_excel('Data2.xlsx')

# Разделение данных на независимые и зависимые переменные

X = data.drop('Готовность остаться', axis=1)  # Независимые переменные

y = data['Готовность остаться']  # Зависимая переменная

# Нормализация данных

scaler = StandardScaler()

X\_scaled = scaler.fit\_transform(X)

# Разделение на тренировочную и тестовую выборки

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_scaled, y, test\_size=0.3, random\_state=42)

# Создание и обучение модели KNN

knn = KNeighborsClassifier(n\_neighbors=5)

knn.fit(X\_train, y\_train)

# Прогнозирование на тестовых данных

y\_pred = knn.predict(X\_test)

# Оценка точности модели

accuracy = accuracy\_score(y\_test, y\_pred)

print(f"Точность модели: {accuracy:.4f}")

# Отчет по классификации

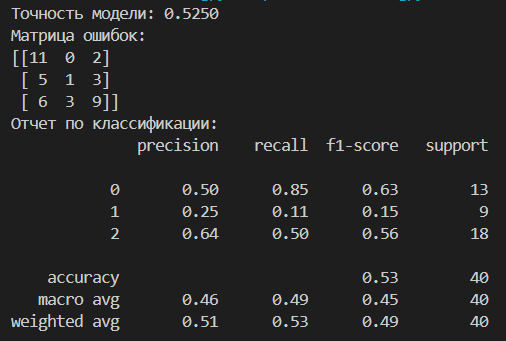
print("Матрица ошибок:")

print(confusion\_matrix(y\_test, y\_pred))

print("Отчет по классификации:")

print(classification\_report(y\_test, y\_pred))

Результат программы:



Здесь классы означают:

* 0 класс – не готовы остаться
* 1 класс – возможно готовы остаться
* 2 класс – готовы остаться

Матрица ошибок – показывает, сколько объектов каждого класса было правильно или ошибочно отнесено к другим классам:

* К 0 классу правильно отнесли 11 человек, 2 человека из 0 класса ошибочно были отнесены во 2 класс
* К 1 классу правильно отнесли 1 человека, 5 человек из 1 класса ошибочно были отнесены во 0 класс, 3 человека из 1 класса ошибочно были отнесены во 2 класс
* Ко 2 классу правильно отнесли 9 человек, 6 человек из 2 класса ошибочно были отнесены в 0 класс, 3 человека из 2 класса ошибочно были отнесены в 1 класс

Precision – точность – показывает, какой процент объектов, предсказанных как принадлежащие этому классу, действительно относятся к этому классу.

* 50% человек, отнесенных к 0 классу, действительно ему принадлежат
* 25% человек, отнесенных к 1 классу, действительно ему принадлежат
* 64% человек, отнесенных ко 2 классу, действительно ему принадлежат

Recall – полноста – показывает, какой процент объектов этого класса модель смогла правильно классифицировать из всех объектов этого класса.

* 85% человек, принадлежащих к 0 классу, были отнесены в этот класс
* 11% человек, принадлежащих к 1 классу, были отнесены в этот класс
* 50% человек, принадлежащих ко 2 классу, были отнесены в этот класс

f1-score – f1-оценка – среднее гармоническое точности и полноты, которое учитывает как ложные положительные, так и ложные отрицательные классификации

Класс 0: F1 = 0.63 – это средний показатель для класса 0. Он учитывает как точность, так и полноту и дает оценку качества модели для этого класса.

Класс 1: F1 = 0.15 – это очень низкий показатель для класса 1, что означает, что модель плохо классифицирует этот класс.

Класс 2: F1 = 0.56 – это средний показатель для класса 2, который также говорит о некоторых проблемах в классификации этого класса.

Macro avg – средняя точность по классам – усреднение метрик по всем классам без учета их пропорций в данных.

* Средняя точность – 0,46
* Средняя полнота – 0,49
* Средняя f1-оценка – 0,45

Weighted average – это усреднение с учетом количества объектов в каждом классе

* Средняя точность – 0,51
* Средняя полнота – 0,53
* Средняя f1-оценка – 0,49

Точность модели – это доля правильных предсказаний на тестовой выборке. Точность данной модели – 0,525, т. е. лишь 52,5% человек были классифицированы правильно. Следовательно, точность модели довольно низкая.

Вывод

В ходе выполнения лабораторной работы был выполнен анализ данных различными методами – с помощью построения регрессии, а также кластеризации и классификации с использованием метода KNN.