Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления

КАФЕДРА Теоретическая информатика и компьютерные технологии

Лабораторная работа

по курсу «Моделирование»

«Модели машинного обучения (логистическая регрессия)»

Студент группы ИУ9-82Б Виленский С. Д.

Преподаватель Домрачева А. Б.

Москва, 2025 г.

ЦЕЛЬ

Приобретение навыка решения задач машинного обучения и построения моделей машинного обучения.

ПОСТАНОВКА ЗАДАЧИ

Провести предобработку предоставленных данных. Построить две предсказательные модели, основанные на библиотечном и собственном методах линейной регрессии, для выбранной предметной области и проверить их эффективность на предобработанных данных.

ИЗУЧЕНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

Предметной областью и объектами исследования являются женщины с различными параметрами функционирования организма и целевым бинарным параметром Outcome, значение которого требуется предсказать. Множество признаков включает в себя:

- количество беременностей,
- концентрация глюкозы в плазме через 2 часа при пероральном тесте на толерантность к глюкозе,
- диастолическое артериальное давление (мм рт. ст.),
- толщина кожной складки трицепса (мм),
- 2-часовой уровень инсулина в сыворотке (мкЕд/мл),
- индекс массы тела (вес в кг/(рост в м)^2),
- диабет родословная функция (функция, которая оценивает вероятность развития диабета на основе семейного анамнеза),
- возраст (лет).

На выборку этих случаев из более обширной базы данных накладывалось несколько ограничений. В частности, все пациенты здесь — женщины в возрасте не менее 21 года индейского происхождения Пима.

Стоит заметить, что все параметры, кроме количества беременностей, имеют область значений, содержащую строго положительные численные значения, в то время как количество беременностей может быть равно нулю.

ПРЕДОБРАБОТКА ДАННЫХ

Первоочередно предоставленный набор данных был проверен на наличие дубликатов. Повторные записи об объектах не были обнаружены, вследствие чего данные на этом этапе никак не были изменены.

После анализа предоставленного набора данных было выявлено, что среди 768 объектов можно обнаружить невалидные значения, а конкретно нулевые значения, среди следующих: концентрация глюкозы – 5; диастолическое артериальное давление – 35; толщина кожной складки трицепса – 227; уровень инсулина – 374 и индекс массы тела – 11. Отношение размеров кластеров относительно целевого параметра Outcome для выборки равно 1.86, из чего следует отсутствие надобности в дополнении каких-либо кластеров в каждой из выборок.

Для обработки пропусков в данных было принято решение их заполнение некоторыми значениями. Была построена матрица корреляции для признаков выборки (рисунок 1) с целью обнаружения возможных зависимостей между параметрами, позволившими бы сформировать корректный алгоритм заполнения пропусков данных.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
Pregnancies	1.000000	0.129459	0.141282	-0.081672	-0.073535	0.017683	-0.033523	0.544341	0.221898
Glucose	0.129459	1.000000	0.152590	0.057328	0.331357	0.221071	0.137337	0.263514	0.466581
BloodPressure	0.141282	0.152590	1.000000	0.207371	0.088933	0.281805	0.041265	0.239528	0.065068
SkinThickness	-0.081672	0.057328	0.207371	1.000000	0.436783	0.392573	0.183928	-0.113970	0.074752
Insulin	-0.073535	0.331357	0.088933	0.436783	1.000000	0.197859	0.185071	-0.042163	0.130548
BMI	0.017683	0.221071	0.281805	0.392573	0.197859	1.000000	0.140647	0.036242	0.292695
DiabetesPedigreeFunction	-0.033523	0.137337	0.041265	0.183928	0.185071	0.140647	1.000000	0.033561	0.173844
Age	0.544341	0.263514	0.239528	-0.113970	-0.042163	0.036242	0.033561	1.000000	0.238356
Outcome	0.221898	0.466581	0.065068	0.074752	0.130548	0.292695	0.173844	0.238356	1.000000

Рисунок 1 − Матрица корреляции для признаков выборки

По построенной матрице корреляции видно, что наиболее тесная связь наблюдается между возрастом и количеством беременностей, инсулином и толщиной кожного покрова, индексом массы тела и толщиной кожного покрова. Все эти зависимости обуславливаются особенностями строения человеческого организма, не являются хаотичными и могут использоваться при восполнении пропусков значений в соответствующих колонках данных.

Для визуального анализа попарной взаимосвязи данных была построена матрица диаграмм рассеяния для каждой пары параметров выборки (рисунок 3).

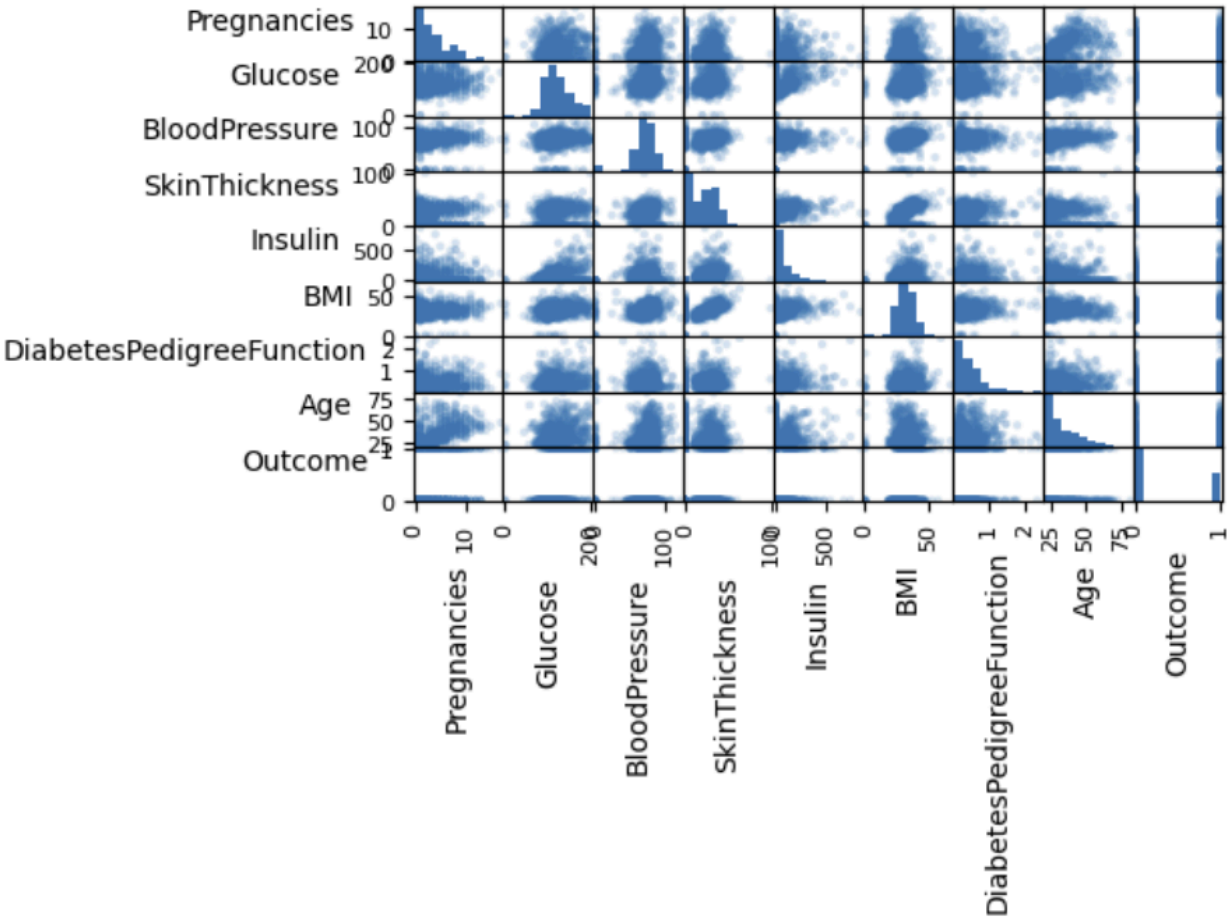

Рисунок 3 − Матрица диаграмм рассеяния данных выборки

По построенной диаграмме рассеяния можно сделать вывод о стохастической зависимости между большинством данных, а также о плотности распределения каждого из параметров, изображенной на диагонали матрицы.

Исходя из проведенного анализа было принято решение о заполнении пропусков средними арифметическими значениями валидных значений соответствующих параметров соседних объектов, так как графики плотностей распределения для всех параметров являются достаточно симметричными, чтобы значение медианы совпадало со средним арифметическим значением.

Данные для каждой из выборок были разделены на обучающую, валидационную и тестовую подвыборки в отношении 6 к 2 к 2 соответственно. Обучающие подвыборки требуются для целевого обучения

моделей, валидационная – для подбора гиперпараметров для моделей и тестовая – для оценки результата работы моделей.

Поскольку в дальнейшем планируется использовать такие методы оптимизации, как логистическая регрессия и метод опорных векторов, была проведена нормализация данных всех подвыборок с использованием данных о диапазонах значений параметров из предметной области:

- количество беременностей: от 0 до 20;
- концентрация глюкозы в плазме через 2 часа при пероральном тесте на толерантность к глюкозе: от 40 до 250;
- диастолическое артериальное давление: от 10 мм рт. ст. до 150 мм рт. ст.;
- толщина кожной складки трицепса: от 4 мм до 120 мм;
- 2-часовой уровень инсулина в сыворотке: от 10 мкЕд/мл до 900 мкЕд/мл;
- индекс массы тела (вес в кг/(рост в м)^2): от 10 до 80;
- диабет родословная функция: от 0 до 2.5,
- возраст (лет): от 21 года до 100 лет.

Таким образом данные были подготовлены для решения задачи машинного обучения (листинг 1). Типом задачи является обучение с учителем, так как в данных выборок имеются значения целевой величины, Outcome.

ПОСТРОЕНИЕ ПРЕДСКАЗАТЕЛЬНЫХ МОДЕЛЕЙ

Для решения поставленной задачи были построены две предсказательные модели: модель на основе библиотечной линейной регрессии с использованием ядра liblinear, соответствующего простейшему персептрону, реализованному программно в другой модели, и модель на основе линейной регрессии собственной реализации (листинг 2).

ПОСТРОЕНИЕ ОПТИМИЗИРУЮЩЕГО АЛГОРИТМА

Для построенных моделей были выбраны различные оптимизирующие алгоритмы: для модели на основе логистической регрессии – линейный метод оптимизации, предполагающий наличие линейной функциональной зависимости;

для модели на основе собственной реализации логистической регрессии – метод градиентного спуска с использованием сигмоиды в качестве функции активации.

РЕШЕНИЕ ПРОБЛЕМ ПЕРЕОБУЧЕНИЯ И УТЕЧКИ ДАННЫХ

Для избежания переобучения использовался метод ограничения максимального количества итераций обучения и подбор шаба обучения на этапе валидации модели.

Отсутствие утечки данных было обеспечено нормализацией данных и отсутствие дублирований значений целевого параметра среди параметров обучающей выборки.

ОЦЕНКА КАЧЕСТВА ПОЛУЧЕННОГО РЕЗУЛЬТАТА

Для оценки качества работы предсказывающих моделей были выбраны следующие метрики:

- $accuracy = \frac{TP + TN}{TP + FP + TN + FN}$;
- $precision = \frac{TP}{TP + FP}$;
- $recall = \frac{TP}{TP + FN}$;
- $f1 = 2\frac{precision * recall}{precision + recall}$

где определены следующие переменные:

- TP (True Positive) – количество верно предсказанных объектов с меткой 1;
- FP (False Positive) – количество объектов с меткой 1, которым была предсказана метка 0;
- TN (True Negative) – количество верно предсказанных объектов с меткой 0;
- FN (False Negative) – количество объектов с меткой 0, которым была предсказана метка 1.

На таблицах 1 и 2 представлены значения f1-меры по результатам тестирования построенных и отвалидированных предсказывающих моделей.

Таблица 1 − Результаты валидирования построенных моделей

Метод модели	f1-мера
Библиотечная логистическая регрессия	0.6118
Собственная логистическая регрессия	0.6264

Таблица 2 − Результаты тестирования построенных моделей

Метод модели	f1-мера
Библиотечная логистическая регрессия	0.4878
Собственная логистическая регрессия	0.587

Как можно заметить, значения f1-меры для двух моделей после валидации на тестовой выборке находятся в одном интервале с шагом в 0.1 в пользу собственной реализации, что можно считать успешным результатом.

ФРАГМЕНТЫ ИСХОДНОГО КОДА

Программная реализация решения задачи машинного обучения представлена в листингах 1-2.

Листинг 1 − Исходный код программы предобработки данных

```
df = pd.read_csv('diabetes.csv')


df_full_size = len(df)
df = df.drop_duplicates()
print(f'Количество дубликатов в датасете: {df_full_size - len(df)}')
print(f"Отношение размеров кластеров: {sum(df['Outcome'] == 0) / sum(df['Outcome'] == 1)}")
```

```
for column_name in ['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']:
    column_values = df[column_name]
    column_mean_value = column_values[column_values != 0].mean()
            df[column_name] = column_values.mask(column_values == 0).fillna(column_mean_value)

df['Pregnancies'] = df['Pregnancies'] / 20;
df['Glucose'] = (df['Glucose'] - 40) / (250 - 40);
df['BloodPressure'] = (df['BloodPressure'] - 10) / (150 - 10);
df['SkinThickness'] = (df['SkinThickness'] - 4) / (120 - 4);
df['Insulin'] = (df['Insulin'] - 10) / (900 - 10);
df['BMI'] = (df['BMI'] - 10) / (80 - 10);
df['DiabetesPedigreeFunction'] = df['DiabetesPedigreeFunction'] / 2.5;
df['Age'] = (df['Age'] - 21) / (100 - 21);
```

Листинг 2 – Исходный код программной реализации модели на основе логистической регрессии собственной реализации

```
class CustomLogisticRegression:
    def __init__(self, lr, n_iters):
        self.lr = lr
        self.n_iters = n_iters
        self.weights = None
        self.bias = None

    def _sigmoid(self, z):
        z = np.clip(z, -500, 500)
        return 1 / (1 + np.exp(-z))

    def fit(self, X, y):
        n_samples, n_features = X.shape
        self.weights = np.ones(n_features)
        self.bias = 0
```

```
        for _ in range(self.n_iters):
            linear_model = np.dot(X, self.weights) + self.bias
            y_predicted = self._sigmoid(linear_model)

            dw = (1 / n_samples) * np.dot(X.T, (y_predicted - y))
            db = (1 / n_samples) * np.sum(y_predicted - y)

            self.weights -= self.lr * dw
            self.bias -= self.lr * db

    def predict(self, X):
        linear_model = np.dot(X, self.weights) + self.bias
        y_predicted = self._sigmoid(linear_model)
        return [1 if i >= 0.5 else 0 for i in y_predicted]
```

ВЫВОДЫ

В результате выполнения лабораторной работы были предобработаны входные данные, построены, отвалидированы и протестированы две различные реализации предсказывающих моделей на основе логистической регрессии.

После предобработки данных было произведено разделение исходной выборки данных на обучающую, валидационную и тестовую подвыборки и нормализация каждой подвыборки.

По результатам обучения и валидирования двух моделей для каждой из них были подобраны оптимальные параметры обучения и алгоритмы оптимизаций.

По результатам тестирования на основе f1-меры можно сделать вывод о валидности построенных моделей по причине эквивалентности результатов их тестирования с погрешностью в 0.1. F1-мера подходит для данной предметной области по причине недопустимости игнорирования ни ложных срабатываний, ни пропущенных важных случаев.