Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский Томский политехнический Университет»



Институт

ИЯТШ

01.03.02 «Прикладная математика и информатика»

**Отчет о практике**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Выполнил:** |  | | | | |
| студент группы | 0В91 |  | Змитрович Н.Л. |  |
|  |  |  |  |  |
| **Проверил:** | Чумаков А.Г. | | | | |
| преподаватель |  |  |  |  |  |
|  |  |  |  |  |  |

Томск – 2021

**Задание**

Используя данные и их описание с источника <https://www.kaggle.com/uciml/pima-indians-diabetes-database> сделайте анализ и обучите модель предсказывать наличие диабета.

**Ход работы**

Для выполнения работы скачаем данные с сайта kaggle и создадим папку data в директории с jupyter notebook. Инициализируем библиотеки pandas для работы с данными и matplotlib, для построения графиков, а также применим функцию самого jupyter notebook %matplotlib notebook, для того, чтобы в дальнейшем можно было изменять размеры графиков. После чего средствами pandas импортируем базу данных и отобразим несколько первых элементов, для визуального понимания структуры данных.

Имеет восемь различных независимых переменных и одну зависимую – **Outcome.** Если значение Outcome пациента равно один, значит у него обнаружен диабет, если ноль, значит у пациента нет диабета. С помощью метода describe проверим основные параметры каждой из переменных, а также узнаем количество пациентов. При отображении первых нескольких строк таблицы, мы можем увидеть, что в таблице нет неопределенных значений, но для того, чтобы убедиться в этом применим комбинацию методов - isnull() и sum(). Метод isnull() вернет таблицу у которой этот метод вызван, только теперь все пустые ячейки (значение в которых равно null) будут иметь значение False, а непустые значение True. Так как в языке Python значение True при суммировании эквивалентно единице, False нулю, то после суммирования по каждой переменной, получим количество пустых ячеек в столбцах. По результату можем сделать вывод, что пустых ячеек в таблице нет.

Также при первичном ознакомлении с базой данных можно заметить что значение Insulin и SkinThicknes принимают значение 0, чего быть не может, значит значения просто пропущены. Построим гистограммы частот для каждой переменной, так как графиков много, чтобы они не наслаивались друг на друга и были легко читаемы в методе hist() передадим параметр figsize=(9,9), чтобы увеличить высоту и ширину основной фигуры в которой будут находится десять графиков. Действительно, по гистограммам частот параметров Insulin и SkinThicknes видим, что имеется достаточно большое количество нулевых значений. Попробуем посчитать это количество для каждого столбца, исключая параметры в которых нулевые значения теоретически могут присутствовать: Pregnancies (количество беременностей), Age (возраст) и Outcome (значения в котором в принципе могут быть равны либо ноль, либо один). В результате получаем, что у всех параметров, кроме DiabetesPedigreeFunction, имеются пропущенные значения, попробуем их заменить. По гистограммам частот можно сделать вывод, что параметры Glucose, BloodPressure и BMI имеют распределения похожие на нормальное, а остальные параметры, кроме Outcome, распределения похожие на параметрическое, значит заменять пропущенные значения на среднее было бы не очень корректно, так как среднее у параметрического распределение сильно смещено от максимального значения, а нам бы хотелось заменить пустые значения на такие, которые имеются у большинства, но при этом опять же из-за того, что некоторые параметры распределены экспоненциально, заменять пропущенные значения на моду, было бы так-же не совсем корректно, поэтому заменим их на медиану. Такая замена, как мне кажется не окажет критического влияния на обучение модели, и не вызовет переобучение на так называемых шумах, которыми по сути и будут являться пустые значения. Сделаем замечание, у параметра SkinThicknes и Insulin, количество пропущенных значений крайне велико относительно всех значений (30% для SkinThicknes и 49% для Insulin).

Перед тем, как производить замену нулевых значений, сохраним индексы элементов с ненулевыми Insulin SkinThicknes. Теперь найдем медианные значения для каждого из параметров, который будем изменять. Важное замечание, при поиске медианы важно исключить пустые значения, так как они могут сильно сместить положение медианы в распределении. Так например для инсулина, если не исключать пропущенные значения, медиана будет равна 30.5, а если исключать, то уже 125. Теперь создадим список с «фичами» нашей базы данных и пройдемся по каждому столбцу, кроме Pregnancies, Age и Outcome (первого и двух последних), заменяем нулевые значения на значение медианы. Медиану будем искать на каждой итерации цикла с помощью маски, которая исключает нулевые значения, после чего применим метод median() и возьмем только текущий столбец. Проверим результат и увидим, что теперь пропущенных (нулевых) значений в таблице нет.

С помощью метода corr() посмотрим на значения корреляции между каждым из параметров. Некоторые параметры имеют достаточно большую корреляцию, например, значение корреляции между возрастом и количеством беременностей достаточно высока (0.544) или глюкозой и инсулином (0.419), значит не стоит использовать регрессионную модель, так как в такой модели нежелательно иметь высокую корреляцию между независимыми переменными. Также заметим, что корреляции SkinThicknes Outcome и Insulin Outcome, достаточно небольшие, а учитывая, что почти треть стобца SkinThicknes и половина столбца Insulin были пропущены, можно считать эти столбцы недостаточно значимыми, для обучения модели и просто не включать в обучающие данные.

Попробуем посмотреть корреляцию между SkinThicknes, Insulin и Outcome в изначальной базе данных, при этом исключая нулевые значения. Для этого воспользуемся двумя, сохраненными до этого списками, с элементами в которых не пропущены SkinThicknes и Insulin, Получаем значения корреляции между SkinThicknes и Outcome равное 0.303 и значения корреляции между Insulin и Outcome равное 0.258. Учитывая, что корреляция между SkinThicknes и Outcome и Insulin и Outcome, в таблице с заменой пропущенных значений соответственно равны 0.215 и 0.204. То есть добавив в данные 30% у SkinThicknes и почти 50% у Insulin медианных значений, мы достаточно сильно ухудшаем корреляцию.

Также учитывать специфику обучения модели методом решающих деревьев. Может получиться так, что пустые значения, которые мы заменили на медиану будут играть большое значение при обучении, так как дерево определяет некоторый барьер, и если значение меньше барьера, то оно будет определено в один лист, а если больше, то в другой. Таким образом мы рискуем определить значительное количество элементов в неправильный лист (так как изначально, значение было пропущено). Поэтому мне кажется, из базы данных стоит вычленить некоторых валидационный датасет, на котором можно проверить будут ли лучше метрики, если обучать модель на данных с заменой пропущенных значений, с заменой пропущенных значений и удалением параметров SkinThicknes и Insulin и с использованием базы данных с исключением пропущенных значений в этих колоннах (только в этих, потому что в остальных количество пропущенных значений мало).

Сначала попробуем обучить модель на таблице с заменой пропущенных значений. Разделим нашу таблицу на 3 части: данные на которых модель будет обучаться, валидационные данные и тестовые данные. Пусть тестовые данные будут состоять из 25% базы данных, остальные определим, как тренировочные, после чего пусть 25% тренировочных данных будут являться валидационными. Так как такое разбиение сложнее, чем то, что может дать модуль skleaern train\_test\_split, то разобьем данные самостоятельно. Для этого инициализируем массив с индексами всей таблицы (числами от 0 до 768), после чего перемешаем его с помощью функции np.random.shuffle, и первые 75% данных запишем в массив train\_indexes, а остальные в test\_indexes. Аналогично разобьем train\_indexes на train\_indexes и validation\_indexes. Можно проверить не потеряли ли мы в процессе разбиения индексы, для этого просуммируем длинны всех трех массивов и получим, что их суммарная длина равна длине первоначального массива (768).

Запишем в таблицу X все параметры из таблицы с заменой пустых значений, кроме результата Outcome, а в таблицу (серию) y только результаты.