Лабораторная работа №13. Обнаружение аномалий.

В лабораторной работе вам необходимо построить модель обнаружения аномалий для определения необычного поведения серверов в центре обработки данных. Теоретический материал для лабораторной работы можно найти в разделе 10 учебного пособия.

1 Загрузка и отображение данных

На первом шаге программы загружается набор данных из файла ex8data1.npy. Из файла загружаются матрицы X, $Xval$ и вектор $yval$.

Матрица X из $m = 307$ строк с $n = 2$ колонками:

$x_1^{(i)}$ – сетевая пропускная способность сервера (мбит/с);

$x_2^{(i)}$ – задержка ответа от сервера (мсек).

Это обучающая выборка для построения модели обнаружения аномалий.

Матрица $Xval$ размером 307×2 – валидационная выборка для настройки параметров алгоритма; вектор $yval$ длиной 307 содержит признаки классов для каждого объекта валидационной выборки: $yval^{(i)} = 0$, если $Xval^{(i)}$ – обычный сервер; $yval^{(i)} = 1$, если $Xval^{(i)}$ имеет аномальное поведение.

Вам необходимо отобразить на плоскости обучающую выборку из матрицы X. Найдите в программе функцию draw_data и дополните ее кодом. Функция принимает на вход матрицу X размером $m \times n$, где m – число объектов в выборке, $n = 2$ – размерность объекта. Каждый объект необходимо отобразить на двумерной плоскости точкой. Пример показан на рис. 1.

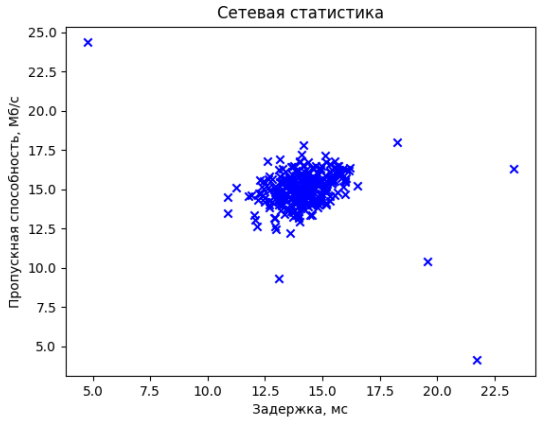

Рис. 1

Удалите оператор return, чтобы перейти к следующему шагу.

2 Оценивание параметров распределения

На данном шаге вам необходимо оценить параметры нормального распределения для каждого признака ($x_1^{(i)}$, $x_2^{(i)}$ и т.д.) обучающей выборки.

Найдите в программе функцию estimate_gaussian и дополните ее необходимым кодом для определения векторов μ и σ^2.

Функция принимает на вход матрицу обучающей выборки X размером $m \times n$, где m – число объектов в выборке, n – число признаков объекта.

Функция должна вернуть:

mu – вектор размером n средних значений (μ) для каждого признака;

sigma2 – вектор размером n дисперсий (σ^2) для каждого признака.

Учтите, что функция должна работать для произвольных размеров m и n.

Когда функция готова, запустите программу. Она вызывает функцию для обучающей выборки X и выводит найденные значения. Если функция реализована правильно, полученные результаты должны совпадать с ожидаемыми.

Удалите оператор return, чтобы перейти к следующему шагу.

3 Модель многомерного нормального распределения

На третьем шаге необходимо построить модель обнаружения аномалий на основе многомерного нормального распределения с использованием параметров, найденных на предыдущем шаге. В данной модели вероятность того, что произвольный объект x является «нормальным», определяется выражением:

$$p(x;\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right),$$

где μ – вектор матожидания (вектор mu в программе); Σ – ковариационная матрица, определяемая как:

$$\Sigma = \begin{bmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \dots & \dots & \dots & 0 \\ 0 & 0 & \dots & \sigma_n^2 \end{bmatrix},$$

где $\sigma_1^2, \dots, \sigma_n^2$ – элементы вектора sigma2 в программе.

Найдите в программе функцию multivariate_gaussian и дополните ее необходимым кодом для вычисления вероятности. Функция принимает параметры:

X – матрица размером $m \times n$ входных объектов, для которых необходимо рассчитать вероятности, m – количество объектов, n – количество признаков;

mu – вектор длиной n матожидания по всем признакам;

sigma2 – вектор длиной n дисперсий по всем признакам.

Функция должна вернуть:

p – вектор длиной m значений вероятности для каждой строки входной матрицы *X*.

Рекомендуется реализация этой функции в матричном виде, чтобы обеспечить высокую производительность. Для превращения вектора *sigma2* в диагональную матрицу можно воспользоваться функцией diag библиотеки NumPy.

Когда функция готова, запустите программу. Она вызывает функцию для валидационной выборки *Xval* и выводит первых пять найденных значений. Если функция реализована правильно, полученные результаты должны совпадать с ожидаемыми.

Удалите оператор return, чтобы перейти к следующему шагу.

4 Выбор порога

Чтобы правильно отличать нормальные объекты от аномальных в системе обнаружения аномалий должен быть настроено значение порога ε с использованием валидационной выборки.

Объект считается аномальным, если:

$$p(x;\mu,\Sigma) < \varepsilon.$$

Для валидационной выборки известны не только признаки объектов *Xval*, но и вектор меток классов *yval*. Значение порога ε подбирается таким образом, чтобы ошибок определения аномалий было как можно меньше.

Поскольку обнаружение аномалий является, как правило, задачей с

несимметричными классами, то качество обнаружения аномалий определяют по мере F_1 или аналогичным. Подробно мера F_1 описана в разделе 6 учебного пособия.

Таким образом, порог ε необходимо подобрать таким, чтобы при нем достигался максимум F_1-меры на валидационной выборке.

Найдите в программе функцию select_threshold и дополните ее необходимым кодом для определения порога. Функция принимает параметры:

yval – вектор меток классов для каждого объекта выборки;

pval – вектор найденных значений вероятностей для каждого объекта выборки.

Функция должна вернуть:

epsilon – найденное оптимальное значение порога;

$F1$ – значение F_1-меры при найденном пороге.

Когда функция готова, запустите программу. Она вызывает функцию для валидационной выборки и выводит полученные значения. Если функция реализована правильно, полученные результаты должны совпадать с ожидаемыми.

Удалите оператор return, чтобы перейти к следующему шагу.

5 Отображение аномальных объектов

На этом шаге нужно отобразить на графике нормальные и аномальные объекты обучающей выборки различными цветами, а также отобразить границу найденного порога ε.

Найдите в программе функцию draw_data_and_fit и дополните ее кодом для вывода графика. Функция принимает параметры:

X – матрица исходных данных размером $m \times n$;

mu – вектор длиной n матожидания по всем признакам;

sigma2 – вектор длиной n дисперсий по всем признакам.

epsilon – значение порога обнаружения аномалий.

Функция не возвращает значений.

Реализуя функцию draw_data_and_fit необходимо вызвать написанную ранее multivariate_gaussian для определения аномальных объектов. Чтобы отобразить границу порога ε можно воспользоваться функцией contour библиотеки PyPlot.

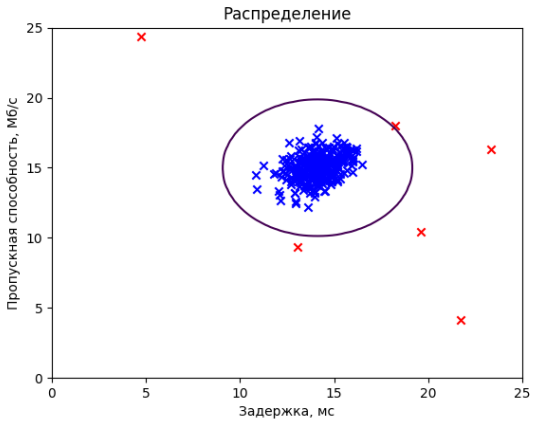

Рис. 2

На рис. 2 показан пример графика, который должен получиться.

Удалите оператор return, чтобы перейти к следующему шагу.

6 Проверка на наборе данных с большим числом параметров

На данном шаге вам не нужно писать своего кода. К данному моменты уже написаны все необходимые функции, и они будут использованы для обнаружения

аномалий на большем наборе данных.

Из файла ex8data2.npy загружается матрица обучающей выборки X размером 1000×11 (т. е. 1000 объектов с 11 признаками), матрица валидационной выборки $Xval$ размером 100×11 и вектор классов валидационной выборки $yval$ длиной 100.

Написанные вами ранее функции используются для обучения модели обнаружения аномалий на этих исходных данных. Вы должна получить 16 аномалий в этом примере.

На этом выполнение лабораторной работы №13 завершается.