

Лабораторная работа 2¹

Цель работы. Освоить понятия булевой функции и логического следствия, применить эти понятия для обработки реальных наборов данных.

Основные понятия

Пусть даны две формулы: φ и ψ , зависящие от переменных x_1, x_2, \dots, x_n . Формулу, зависящую от переменных, будем обозначать через $\varphi(x_1, x_2, \dots, x_n)$. Что значит, что из формулы $\varphi(x_1, \dots, x_n)$ *следует* формула $\psi(x_1, \dots, x_n)$?

Будем говорить о *логическом следствии* формулы ψ из формулы φ и писать

$$\varphi(x_1, \dots, x_n) \models \psi(x_1, \dots, x_n) \quad \text{или} \quad \frac{\varphi(x_1, \dots, x_n)}{\psi(x_1, \dots, x_n)},$$

если истинность левой части гарантирует истинность правой части. Итак, логическое следствие выполняется, если для всякого набора значений переменных x_1, \dots, x_n , при которых $\varphi(x_1, \dots, x_n) = 1$, справедливо и $\psi(x_1, \dots, x_n) = 1$. И логическое следствие нарушается, если существует хотя бы один набор значений переменных x_1, \dots, x_n , при котором $\varphi(x_1, \dots, x_n) = 1$, но $\psi(x_1, \dots, x_n) = 0$.

Булевой функцией от переменных x_1, x_2, \dots, x_n называется таблица соответствия, указывающая для каждого из возможных наборов значений пропозициональных переменных x_1, x_2, \dots, x_n соответствующее этому набору значение 0 или 1.

Задания на лабораторную работу

1. В качестве исходных данных вам даны два набора данных: а) результаты автоматического распознавания (методами машинного обучения) результатов обследования пациентов методом однопротонной эмиссионной томографии — для 267 наблюдений выделено 22 входных признака и один выходной (0 — заболевание не выявлено, 1 — выявлено); б) статистика по мошенничеству (10 бинарных признаков, ещё один количественный признак, который потребуется исключить, и один выходной бинарный признак). Ознакомьтесь с этими данными и запустите заготовку на языке Python, которая должна их прочитать.
2. Переберите все возможные пары входных признаков и определите, какие комбинации значений выбранных двух признаков не встречаются в выборке. Аналогичным образом переберите все возможные тройки входных признаков и определите, какие комбинации значений выбранных трех признаков не встречаются в выборке. Сформулируйте результат анализа выборки на логическом языке, используя понятие логического следствия.

¹Математическая логика и теория алгоритмов (Гренкин Г.В., ВВГУ, 2023)

3. Выясните, какие сочетания значения какого-нибудь входного признака и значения выходного признака не встречаются в выборке. Сделайте содержательный вывод. По аналогии выясните, какие сочетания значений двух входных признаков и значения выходного признака не встречаются в выборке.
4. Проведите вычисления для обоих файлов: `spect.tsv` и `analcata_data_fraud.tsv`. Во втором из них не забудьте убрать столбец `Total` с количественным признаком, чтобы оставить только бинарные признаки.

Рекомендации по выполнению лабораторной работы

Воспользуйтесь приложенной заготовкой:

```
import csv

with open("spect.tsv") as file:
    tsv_file = csv.reader(file, delimiter="\t")
    header = next(tsv_file, None) # пропустить заголовки
    data_str = [row for row in tsv_file]
# список data_str содержит списки значений признаков в каждой строке

data_size = len(data_str) # число наблюдений
num_cols = len(data_str[0]) # число столбцов (общее число бинарных признаков)

# data_x - список списков из атрибутов, расположенных в каждой строке файла
# data_y - список значений выходных признаков для каждой строки
data_x = [[] for _ in range(data_size)]
data_y = [0.0 for _ in range(data_size)]
for i in range(data_size):
    # применяем функцию int к каждому элементу списка
    data_x[i] = list(map(int, data_str[i]))
    data_y[i] = data_x[i].pop() # удаляем из списка data_x[i] последний элемент
x_len = num_cols - 1 # число входных признаков
x_names = header[:-1] # имена входных признаков

print("Наблюдений:", data_size)
print("Входных признаков:", x_len)
print(x_names)

#####
```

```

for i in range(x_len):
    for j in range(i + 1, x_len):
        for A in 0, 1:
            for B in 0, 1:
                # считаем, сколько раз i-й признак равен A и j-й признак равен B
                cnt = 0
                for k in range(data_size):
                    if data_x[k][i] == A and data_x[k][j] == B:
                        cnt += 1
                # если такая комбинация не встретилась ни разу, выводим
                if cnt == 0:
                    print(x_names[i], "=", A, ",", x_names[j], "=", B)

```

Реализуйте аналогичные вычисления для троек признаков

```

for i in range(x_len):
    for j in range(i + 1, x_len):
        for k in range(j + 1, x_len):
            for A in 0, 1:
                for B in 0, 1:
                    for C in 0, 1:
                        # считаем, сколько раз i-й признак равен A и
                        # j-й признак равен B и k-й признак равен C
                        cnt = 0
                        # вставьте ваш код сюда (~ 3 строчки)

                        # если такая комбинация не встретилась ни разу, выводим
                        if cnt == 0:
                            print(x_names[i], "=", A, ",", x_names[j], "=", B, \
                                ",", x_names[k], "=", C)

```

Реализуйте аналогичные вычисления для пар и троек,

в которые входит выходной признак

```

for i in range(x_len):
    for A in 0, 1:

```

```

for B in 0, 1:
    # считаем, сколько раз i-й признак равен A и выходной признак равен B
    cnt = 0
    # вставьте ваш код сюда (~ 3 строчки)

    # если такая комбинация не встретилась ни разу, выводим
    if cnt == 0:
        print(x_names[i], "=", A, ",", "Y", "=", B)

for i in range(x_len):
    for j in range(i + 1, x_len):
        for A in 0, 1:
            for B in 0, 1:
                for C in 0, 1:
                    # считаем, сколько раз i-й признак равен A и j-й признак равен B
                    # и выходной признак равен C
                    cnt = 0
                    # вставьте ваш код сюда (~ 3 строчки)

                    # если такая комбинация не встретилась ни разу, выводим
                    if cnt == 0:
                        print(x_names[i], "=", A, ",", x_names[j], "=", B, \
                            ",", "Y", "=", C)

```