

Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И  
РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления

Кафедра Интеллектуальных информационных технологий

**Отчет по лабораторной работе №3**

**“Байесовский классификатор”**

по дисциплине

Статические основы индукционного вывода

Выполнил:

Якимович И.В.

Студент группы 121703

Проверил:

Русина Н.В.

Минск, 2023

## 1. Цели

Приобрести навыки построения байесовского классификатора

## 2. Задачи

1. Построить модель гауссовского наивного байесовского классификатора для бинарной классификации
2. Оценить точность модели

### Ход выполнения:

1. Импортирование необходимых библиотек:

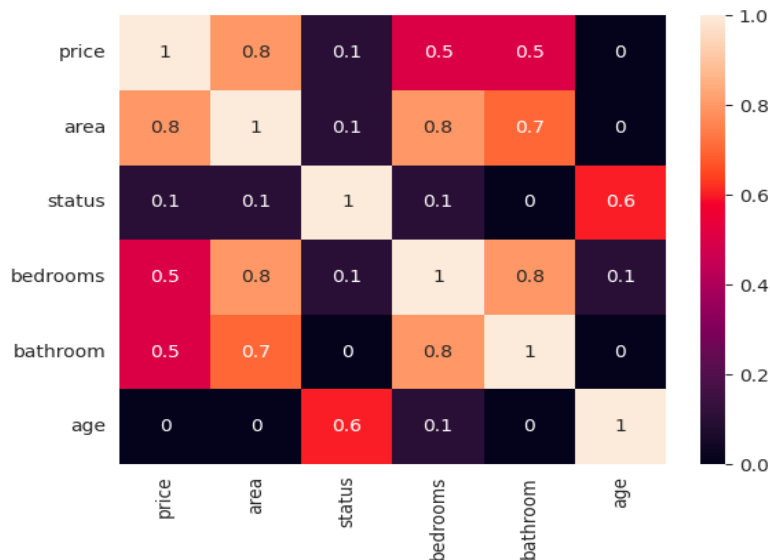
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns

from sklearn import metrics
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
```

2. Создание карты корреляции

```
sns.heatmap(
    round(
        abs(dataset.corr()),
        1,
    ),
    annot=True,
)
```

<Axes: >



Создание карты корреляции между признаками с использованием `sns.heatmap()`. Это позволяет визуализировать взаимосвязь между признаками и определить степень корреляции.

### 3. Построение модели

```
train_input, test_input, train_output, test_output = train_test_split(
    dataset[["area", "age"]],
    (dataset["price"] > 100).astype(int),
    test_size=0.2
)
```

```
model = GaussianNB()
model.fit(train_input, train_output)

predictions = model.predict(test_input)
accuracy = metrics.accuracy_score(predictions, test_output)

print(f"Точность модели на тестовом участке = {accuracy}")
```

Точность модели на тестовом участке = 0.8620689655172413

1. Используя функцию `train_test_split`, данные из `dataset` разделяются на обучающие и тестовые наборы. `dataset[["area", "age"]]` представляет входные признаки, которые состоят из столбцов 'area'

и 'age'. `(dataset["price"] > 100).astype(int)` представляет целевую переменную, где значение 'price' больше 100 преобразуется в 1, а остальные значения в 0. `test_size=0.2` указывает, что тестовый набор данных должен составлять 20% от общего размера данных.

2. Создание модели гауссовского наивного байесовского классификатора `GaussianNB()`.
3. Модель обучается на обучающем наборе данных с помощью метода `fit(train_input, train_output)`.
4. Выполняется прогнозирование на тестовом наборе данных с помощью метода `predict(test_input)`.
5. Вычисляется точность модели на тестовом наборе данных с использованием функции `accuracy_score`. Она сравнивает предсказанные значения `predictions` с фактическими значениями `test_output` и вычисляет долю правильных прогнозов.
6. Значение точности модели выводится на экран с помощью `print(f"Точность модели на тестовом участке = {accuracy}").`