



Министерство науки и высшего образования Российской Федерации  
Калужский филиал федерального государственного автономного  
образовательного учреждения высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(КФ МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

## ЛАБОРАТОРНАЯ РАБОТА

### «Построение нейронной сети для прогнозирования стоимости квартир»

по дисциплине: «Методы машинного обучения»

Выполнил: студент группы ИУК4-72Б

(Подпись)

Е.В. Губин

(И.О. Фамилия)

Проверил(-а):

(Подпись)

М. Г. Семененко

(И.О. Фамилия)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Калуга, 2025

**Целью** выполнения работы является формирование навыков создания наборов данных для обучения нейронных сетей.

**Задание:** создать синтетический набор данных для обучения сети.

### Листинг программы:

```
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

from tensorflow import keras
from tensorflow.keras import layers

def generate_housing_data(n_samples=1000, random_state=42):
    np.random.seed(random_state)

    size = np.random.normal(150, 50, n_samples)           # площадь, м2
    bedrooms = np.random.randint(1, 6, n_samples)         # число спален
    bathrooms = np.random.randint(1, 4, n_samples)        # число ванн
    location_score = np.random.uniform(1, 10, n_samples)  # «качество» района
    age = np.random.exponential(20, n_samples)            # возраст дома, лет

    base_price = 100000
    price = (
        base_price
        + size * 2000
        + bedrooms * 50000
        + bathrooms * 30000
        + location_score * 15000
        - age * 3000
        + size * location_score * 100
        + np.random.normal(0, 50000, n_samples) # шум
    )

    data = pd.DataFrame(
        {
            "size": size,
            "bedrooms": bedrooms,
            "bathrooms": bathrooms,
            "location_score": location_score,
            "age": age,
            "price": price,
        }
    )
    return data

def visualize_housing_data(data):
    fig, axes = plt.subplots(2, 2, figsize=(15, 10))

    axes[0, 0].scatter(data["size"], data["price"], alpha=0.6)
    axes[0, 0].set_xlabel("Площадь (м²)")
    axes[0, 0].set_ylabel("Цена")
    axes[0, 0].set_title("Цена и площадь")

    bedroom_price = data.groupby("bedrooms")["price"].mean()
```

```

axes[0, 1].bar(bedroom_price.index, bedroom_price.values)
axes[0, 1].set_xlabel("Спальни")
axes[0, 1].set_ylabel("Средняя цена")
axes[0, 1].set_title("Цена и кол-во спален")

axes[1, 0].hist(data["price"], bins=50, alpha=0.7)
axes[1, 0].set_xlabel("Цена")
axes[1, 0].set_ylabel("Частота")
axes[1, 0].set_title("Распределение цен")

axes[1, 1].scatter(data["location_score"], data["price"], alpha=0.6)
axes[1, 1].set_xlabel("Оценка локации")
axes[1, 1].set_ylabel("Цена")
axes[1, 1].set_title("Цена и локация")

plt.tight_layout()
plt.show()

def main():
    housing_data = generate_housing_data(2000)
    print(housing_data.head())
    print(f"\nРазмер датасета: {housing_data.shape}")

    visualize_housing_data(housing_data)

```

## Результат выполнения работы:

Таблица №1 – Первые 5 записей созданной выборки

	Площадь	Комнаты	Уборные	Оценка расположения	Возраст	Цена
0	174.84	3	3	9.235886	32.95	89195,90
1	143.09	3	3	2.407259	2.32	57590,04
2	182.38	3	3	7.766183	10.12	92634,08
3	226.15	4	3	6.114418	8.22	102901,60
4	138.29	5	2	3.471993	16.86	76070,90

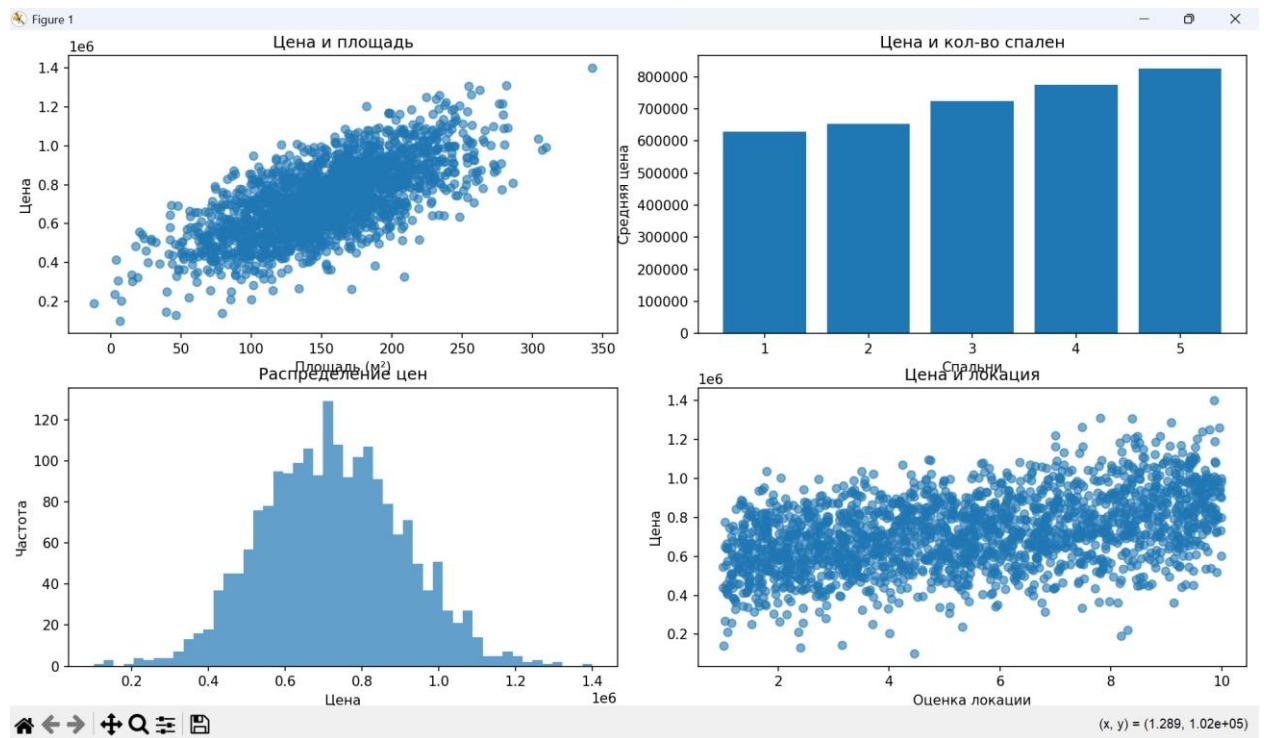


Рис.1 «Визуализация сгенерированных данных»

**Вывод:** в ходе выполнения лабораторной работы были сформированы практические навыки создания наборов данных для обучения нейронных сетей.