**Лабораторная работа № 2**

**ВИЗУАЛИЗАЦИЯ ДАННЫХ**

**Цель работы:**изучение программных средств для визуализации наборов данных.

**Задача**:установка и настройка matplotlib, seaborn; изучение основных типов графиков библиотеки matplotlib; изучение основных типов графиков библиотеки seaborn; получение навыков анализа данных по визуальным представлениям данных

Рассмотрим основные признаки, представленный в наборе. Загрузим набор данных с использованием pandas и выведем признаки набора данных (рисунок 1).

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1 – Загрузка данных получение и первичный анализ признаков

Для детального изучения воспользуемся методом info() класса DataFrame (рисунок 2).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 – Информация о признаках набора данных

Для представления распределения простого количественного признака подходит обычная гистограмма, содержащаяся во всех библиотеках (рисунок 3).

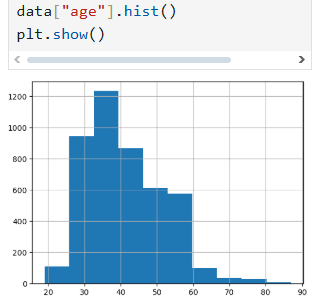


Рисунок 3 — Информация о признаках набора данных

Один из эффективных типов графиков для анализа количественных признаков – это «ящик с усами» (boxplot). На рисунке 4 показан код и реализованный график.

Изображение выглядит как текст, снимок экрана, число, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4 — График «ящик с усами» для отдельного признака

Типичным категориальным признаком в анализируемом наборе данных является «Работа» или «Образование». Под категориальный признак подходит также «Подписка» (хотя он является логическим). На рисунке 5 и 6 представлены графики типа countplot() из библиотеки seaborn, которые строят гистограммы, но не по сырым данным, а по рассчитанному количеству разных значений признака.

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 5 – Визуализация категории «Работа»

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 6 – Визуализация распределения признака «Подписка»

Рассмотрим пример, демонстрирующий сравнение распределений показателей, связанных с возрастом, балансом и длительностью разговора с клиентом. На рисунке 7 представлен код для отбора требуемых показателей.

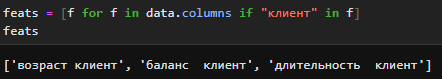


Рисунок 7 — Отбор показателей, связанных с займами клиентов

Часто используют попарное сравнение признаков для обеспечения широкого взгляда на набор данных (рисунок 8). На диагональных графиках рисунка 8 представлены гистограммы распределения отдельного признака, на внедиагональных позициях – попарные распределения.



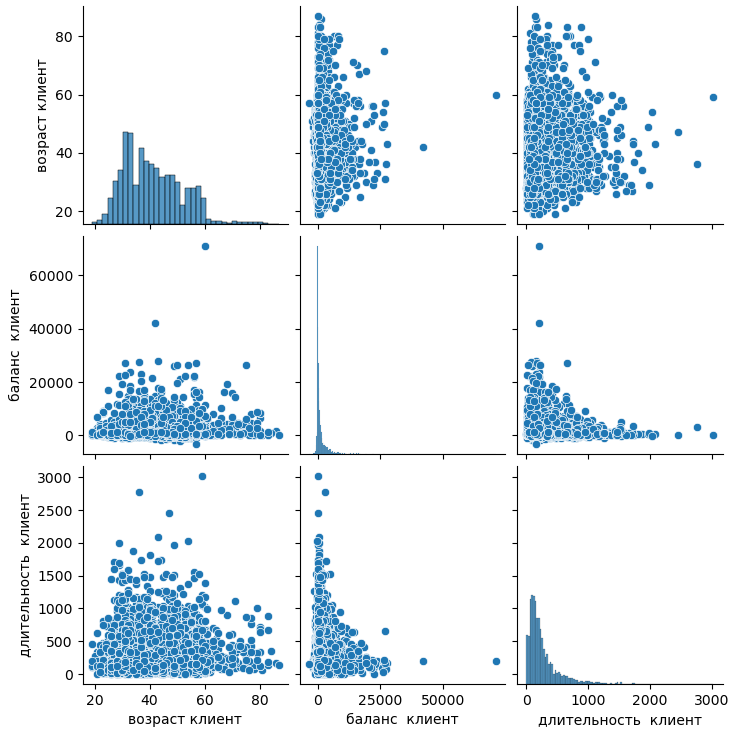


Рисунок 8 — Попарные распределения признаков

Можно реализовать более сложные графики. Например, если требуется добавить к существующим признакам, целевой признак «подписка» и раскрасить разные типы элементов, то можно воспользоваться попарными распределениями, но с отображением подмножеств отказов (рисунок 9).

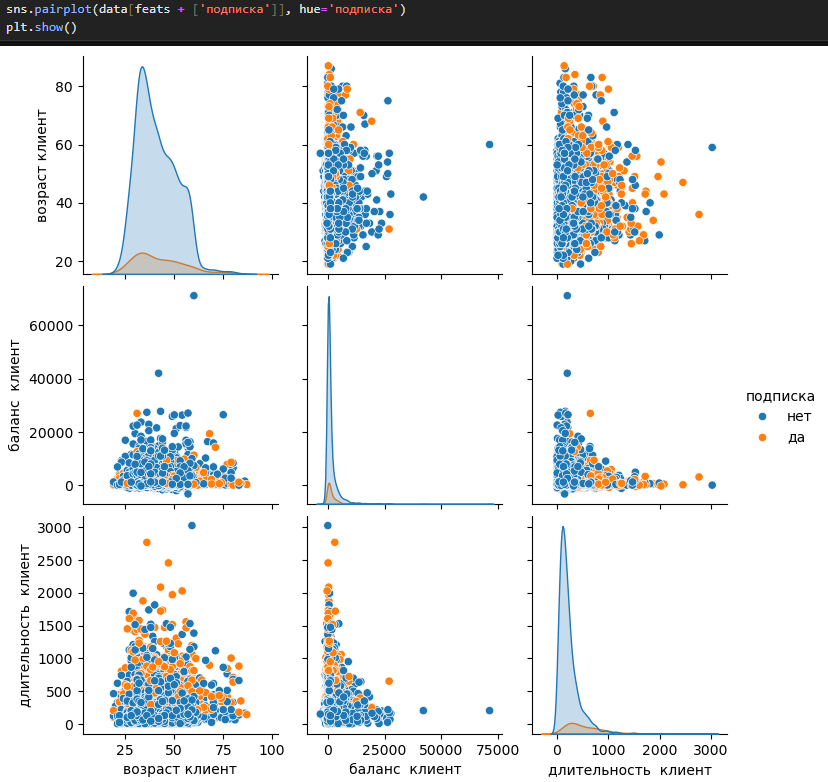


Рисунок 9 – Попарные распределения признаков с визуализацией подписок

На рисунке 10 показан пример использования графика scatter библиотеки matplotlib, предназначенного для вывода множества точек с более тонкой настройки параметров графика.

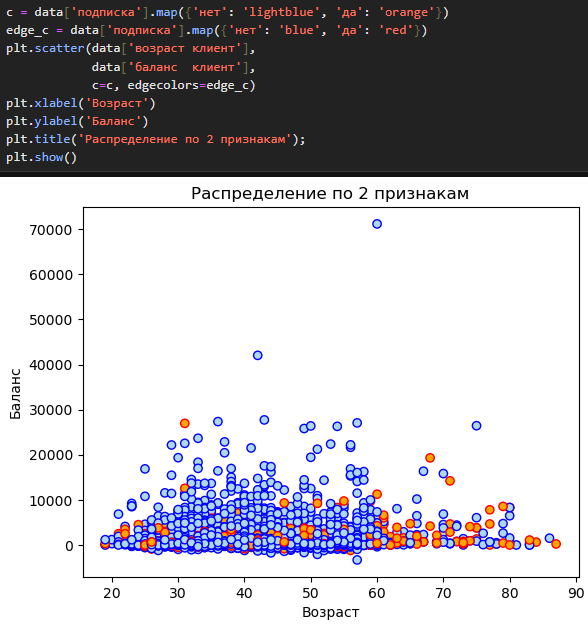


Рисунок 10 – Распределение по 2 признакам с визуализацией подписок

В реальных задачах машинного обучения при первичном анализе данных необходимо выявить корреляции признаков обучающей выборки. В пакете Pandas имеется встроенный инструмент для этого – метод corr() класса DataFrame. На рисунке 11 показан фрагмент вывода этой функции.

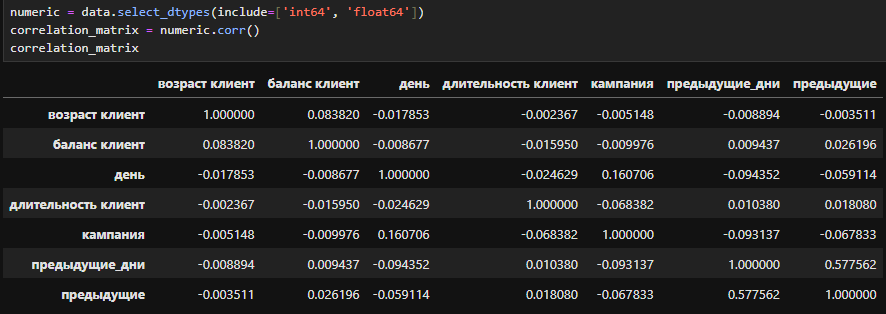


Рисунок 11 – Определение коррелирующих признаков набора данных

Используем специальный тип графика – heatmap для выделения больших коэффициентов корреляции (рисунок 12).

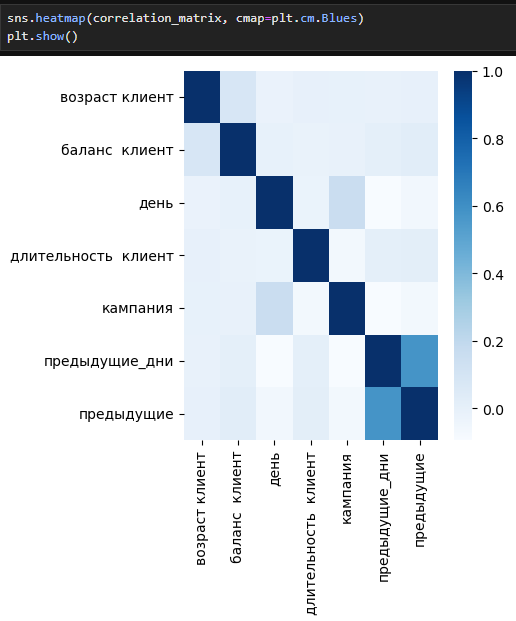


Рисунок 12 – Визуализация матрицы корреляции с использованием графика типа heatmap

**Приложение 1**

import numpy as np

import pandas as pd

from matplotlib import pyplot as plt

import seaborn as sns

%matplotlib inline

data\_path = "D:/py\_anal\_data/bank/bank-ru.csv"

data = pd.read\_csv(data\_path, sep=';')

data.head()

data.info()

data["возраст клиент"].hist()

sns.boxplot(data['баланс клиент'])

sns.countplot(data["работа"])

sns.countplot(data["подписка"])

feats = [f for f in data.columns if "клиент" in f]

sns.pairplot(data[feats + ['подписка']], hue='подписка')

c = data['подписка'].map({'нет': 'lightblue', 'да': 'orange'})

edge\_c = data['подписка'].map({'нет': 'blue', 'да': 'red'})

plt.scatter(data['возраст клиент'],

data['баланс клиент'],

c=c, edgecolors=edge\_c)

plt.xlabel('Возраст')

plt.ylabel('Баланс')

plt.title('Распределение по 2 признакам');

numeric = data.select\_dtypes(include=['int64', 'float64'])

correlation\_matrix = numeric.corr()

correlation\_matrix

sns.heatmap(correlation\_matrix, cmap=plt.cm.Blues)

plt.show()