

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего  
образования  
«Омский государственный технический университет»

Факультет информационных технологий и компьютерных систем  
Кафедра «Прикладная математика и фундаментальная информатика»

**Домашнее задание**

по дисциплине Практикум по программированию

Студента(ки) Григора Александра Антоновича  
фамилия, имя, отчество полностью

Курс 2 Группа ФИТ-222

Направление 02.03.02. Фундаментальная информатика и  
информационные технологии  
код, наименование

Руководитель ст.преподаватель  
должность, ученая степень, звание  
Саматов А. П.  
фамилия, инициалы, дата, подпись

Выполнил \_\_\_\_\_  
дата, подпись студента(ки)

Итоговый рейтинг	
------------------	--

Омск 2023

ВВЕДЕНИЕ .....	3
1.Поиск и загрузка данных .....	4
2.1 Гистограмма распределения числового признака .....	5
2.2 Диаграмма «ящик с усами» числового признака .....	6
2.3 Круговая диаграмма номинативного признака .....	6
2.4 Тепловая карта.....	7
2.5 Диаграмма countplot с группировкой по двум номинативным признакам.....	8
3 Предварительная обработка данных .....	9
ЗАКЛЮЧЕНИЕ .....	11
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	12

## ВВЕДЕНИЕ

В ходе этого курса мы изучили не только сам Python, но и несколько ключевых библиотек, которые позволяют расширить его возможности и применить его в различных областях.

Одной из первых и наиболее важных библиотек, которую мы изучили, была NumPy. Она предоставила нам инструменты для работы с многомерными массивами и матрицами, что позволяет удобно хранить и обрабатывать данные, а также выполнять математические операции.

Затем мы перешли к Matplotlib, который помог нам визуализировать данные. Создание графиков и диаграмм стало проще благодаря этой библиотеке. Мы научились настраивать внешний вид графиков и делать их информативными.

SymPy оказалась весьма полезной для символьных вычислений. Мы научились решать уравнения, дифференцировать и интегрировать функции, а также проводить анализ математических выражений.

SciPy добавила нам множество возможностей для научных и инженерных вычислений. Модули для оптимизации, решения дифференциальных уравнений и статистики были особенно полезны.

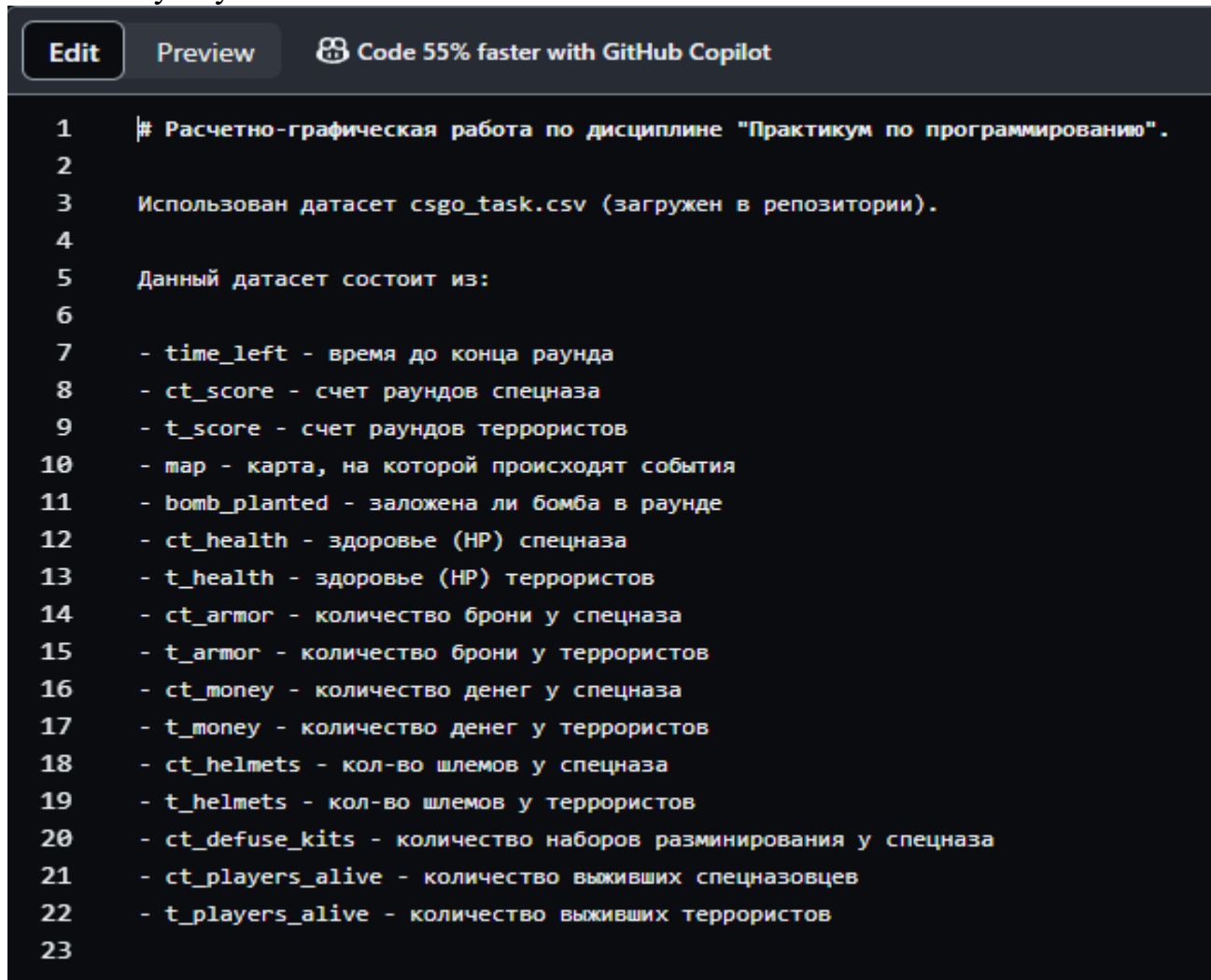
В ходе курса мы также познакомились с библиотекой Pandas, которая позволяет удобно анализировать данные. Создание и манипуляции с DataFrame сделали анализ данных более доступным.

Seaborn стала нашим партнером в создании статистических графиков. Эта библиотека упростила создание гистограмм, «ящиков с усами» и тепловых карт.

Все эти библиотеки расширили наши возможности в программировании на Python и анализе данных. Курс был интересным и познавательным, и я готов применить полученные навыки в своей работе. Я уверен, что эти инструменты станут незаменимыми помощниками в различных проектах.

## 1. Поиск и загрузка данных

Использован датасет `csgo_task.csv`, выданный нашей группе на парах по машинному обучению.



```
1  # Расчетно-графическая работа по дисциплине "Практикум по программированию".
2
3  Использован датасет csgo_task.csv (загружен в репозитории).
4
5  Данный датасет состоит из:
6
7  - time_left - время до конца раунда
8  - ct_score - счет раундов спецназа
9  - t_score - счет раундов террористов
10 - map - карта, на которой происходят события
11 - bomb_planted - заложена ли бомба в раунде
12 - ct_health - здоровье (HP) спецназа
13 - t_health - здоровье (HP) террористов
14 - ct_armor - количество брони у спецназа
15 - t_armor - количество брони у террористов
16 - ct_money - количество денег у спецназа
17 - t_money - количество денег у террористов
18 - ct_helmets - кол-во шлемов у спецназа
19 - t_helmets - кол-во шлемов у террористов
20 - ct_defuse_kits - количество наборов разминирования у спецназа
21 - ct_players_alive - количество выживших спецназовцев
22 - t_players_alive - количество выживших террористов
23
```

Рисунок 1 – файл README.md

Датасет был загружен в ноутбук командой `read_csv()` библиотеки `pandas`.

```
data=pd.read_csv('csgo_task.csv')
```

Рисунок 2 – загрузка датасета

Данный датасет выглядит как набор из 116789 строк и 16 столбцов, разделенных точкой с запятой, в которых описаны раунды в неизвестной игре CS:GO. В нем учитываются многие факторы, такие как установлена ли бомба, сколько выжило игроков в конце раунда, на какой секунде раунд закончился и так далее.

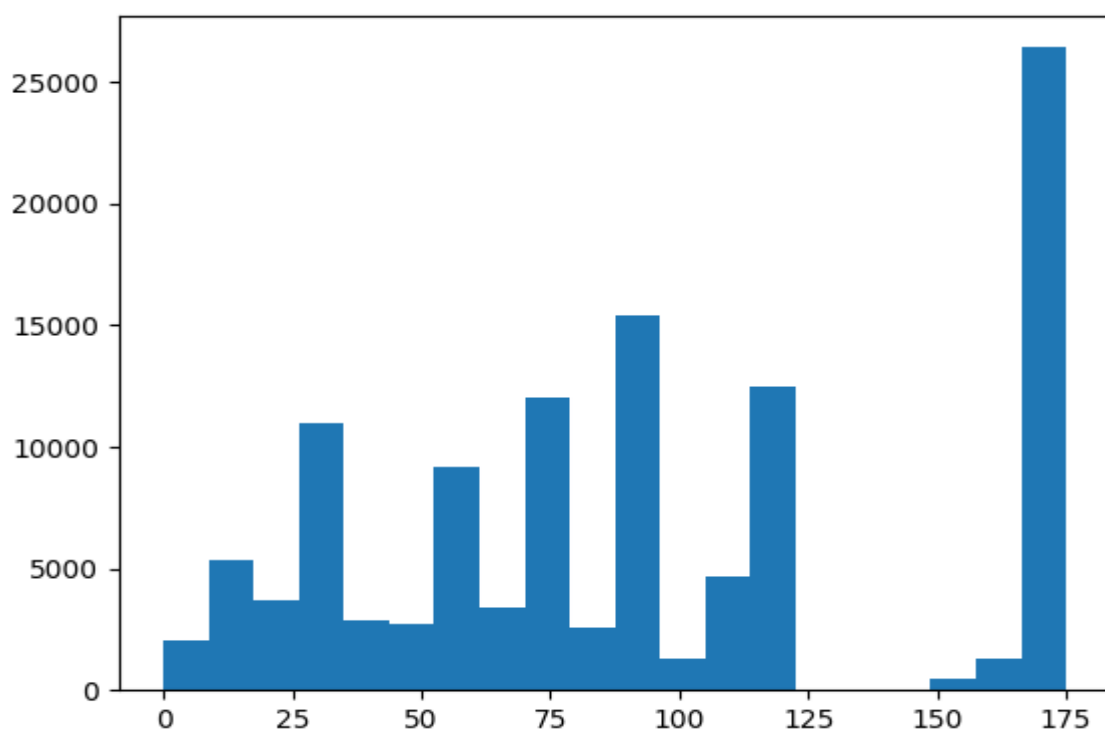
```
# первые 5 строк
data.head(5)
```

	time_left	ct_score	t_score	map	bomb_planted	ct_health	t_health	ct_armor	t_armor	ct_money	t_money	ct_helmets	t_hel
0	175.00	0.0	0.0	de_dust2	False	500.0	500.0	0.0	0.0	4000.0	4000.0	0.0	
1	156.03	0.0	0.0	de_dust2	False	500.0	500.0	400.0	300.0	600.0	650.0	0.0	
2	96.03	0.0	0.0	de_dust2	False	391.0	400.0	294.0	200.0	750.0	500.0	0.0	
3	76.03	0.0	0.0	de_dust2	False	391.0	400.0	294.0	200.0	750.0	500.0	0.0	
4	174.97	1.0	0.0	de_dust2	False	500.0	500.0	192.0	0.0	18350.0	10750.0	0.0	

**Рисунок 3 – небольшая часть датасета, выведенного в виде таблицы**

## 2.1 Гистограмма распределения числового признака

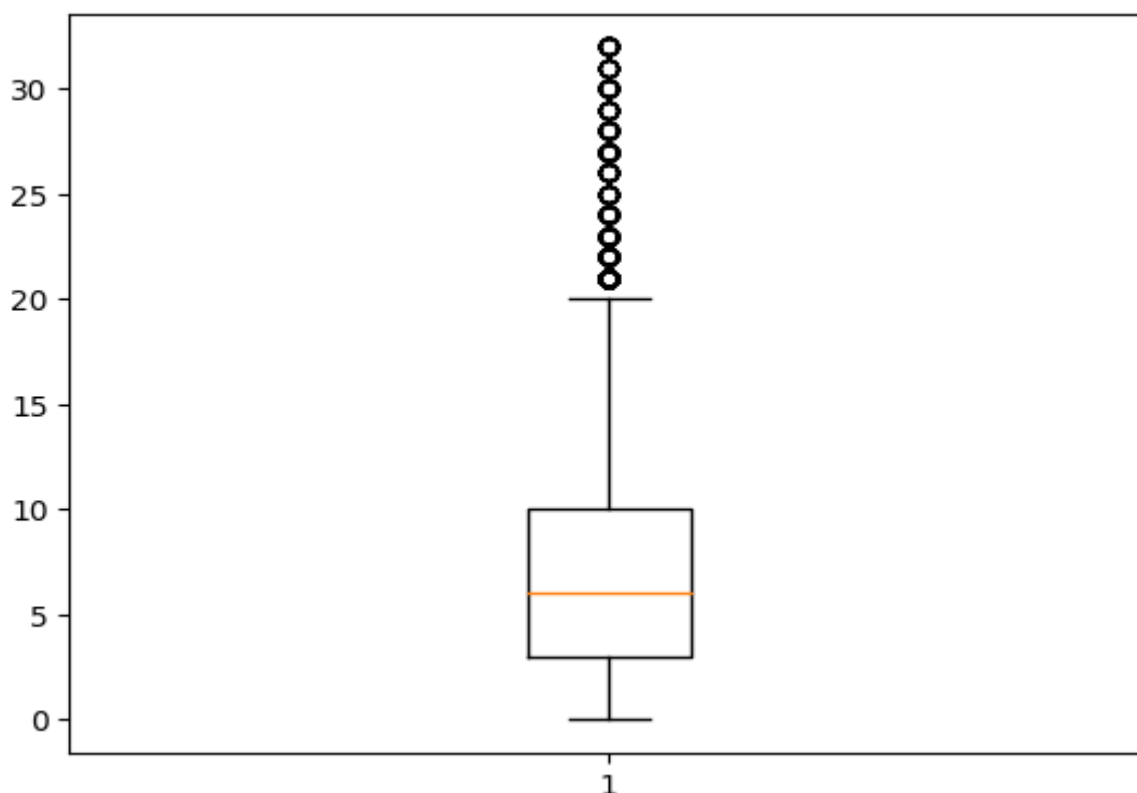
Гистограмма — способ представления табличных данных в графическом виде — в виде столбчатой диаграммы. Количественные соотношения некоторого показателя представлены в виде прямоугольников, площади которых пропорциональны. На гистограмме видно оставшееся время до конца раунда.



**Рисунок 4 – гистограмма столбца time\_left**

## 2.2 Диаграмма «ящик с усами» числового признака

Диаграмма «ящик с усами» — график, использующийся в описательной статистике, компактно изображающий одномерное распределение вероятностей. Такой вид диаграммы в удобной форме показывает медиану (или, если нужно, среднее), нижний и верхний квартили, минимальное и максимальное значение выборки и выбросы. Несколько таких ящиков можно нарисовать бок о бок, чтобы визуально сравнивать одно распределение с другим; их можно располагать как горизонтально, так и вертикально. Расстояния между различными частями ящика позволяют определить степень разброса (дисперсии) и асимметрии данных и выявить выбросы.



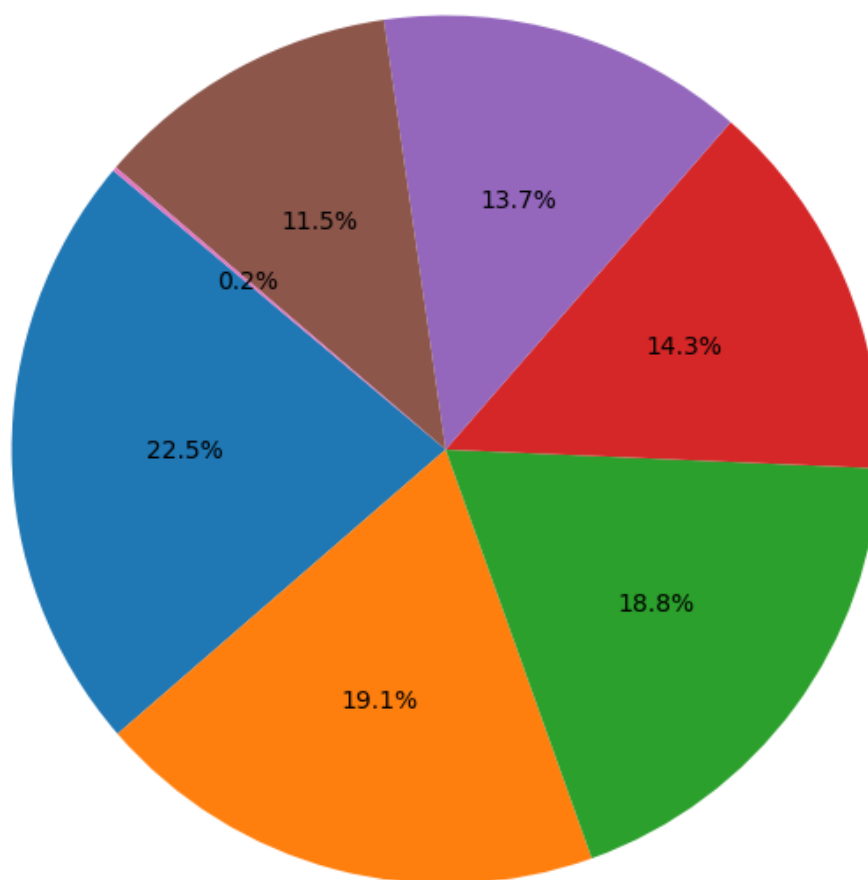
**Рисунок 5 – Диаграмма «ящик с усами» столбца ct\_score**

## 2.3 Круговая диаграмма номинативного признака

Круговая диаграмма — это круговая статистическая диаграмма, которая разделена на срезы, чтобы проиллюстрировать числовую пропорцию. На круговой диаграмме длина дуги каждого среза пропорциональна величине, которую он представляет. На данной круговой диаграмме видно, что

раунды проходили на разных картах в разных объемах.

Карта



**Рисунок 6 – Круговая диаграмма Age**

## 2.4 Тепловая карта

Тепловая карта — графическое представление данных, где индивидуальные значения в таблице отображаются при помощи цвета.

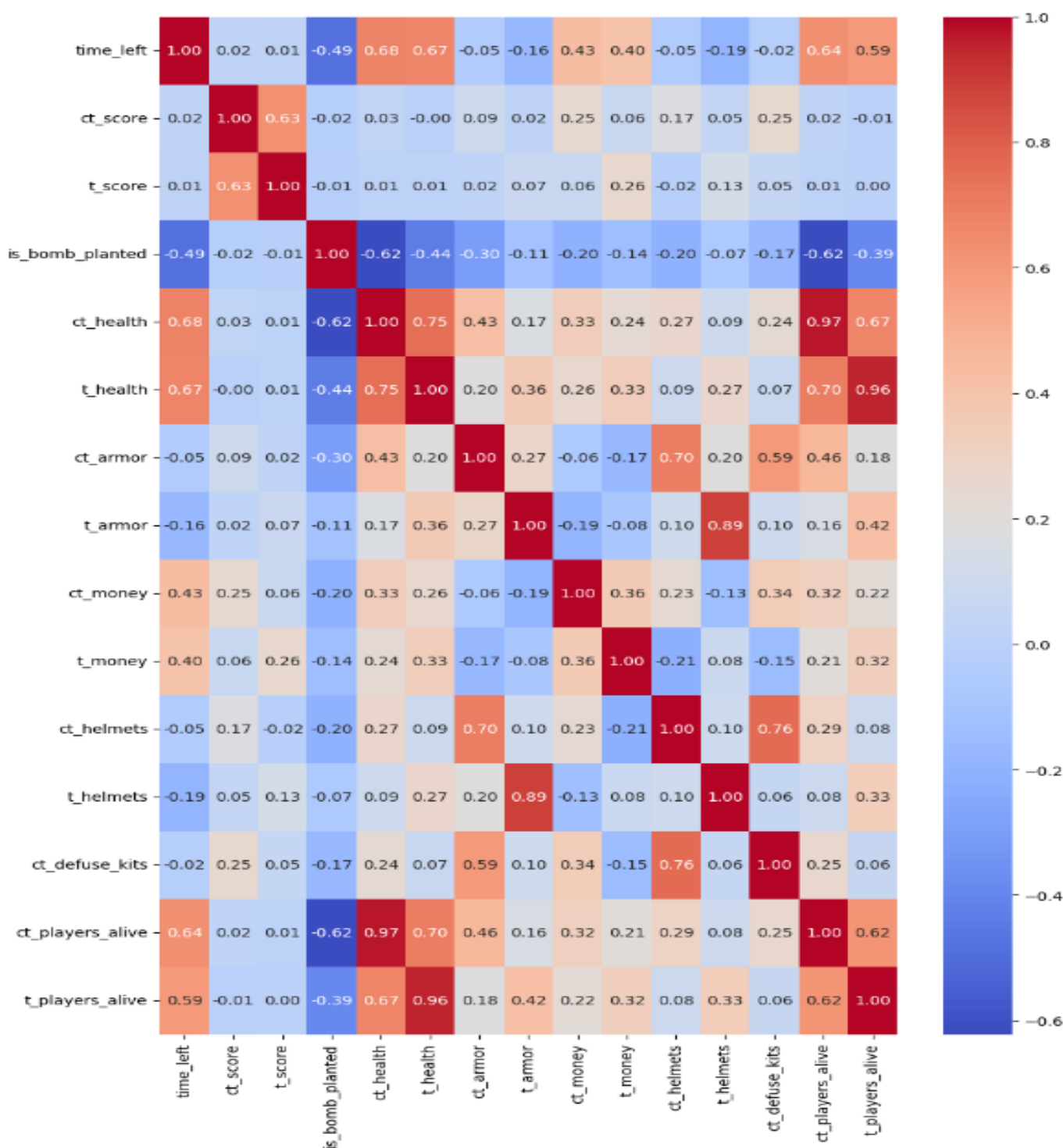
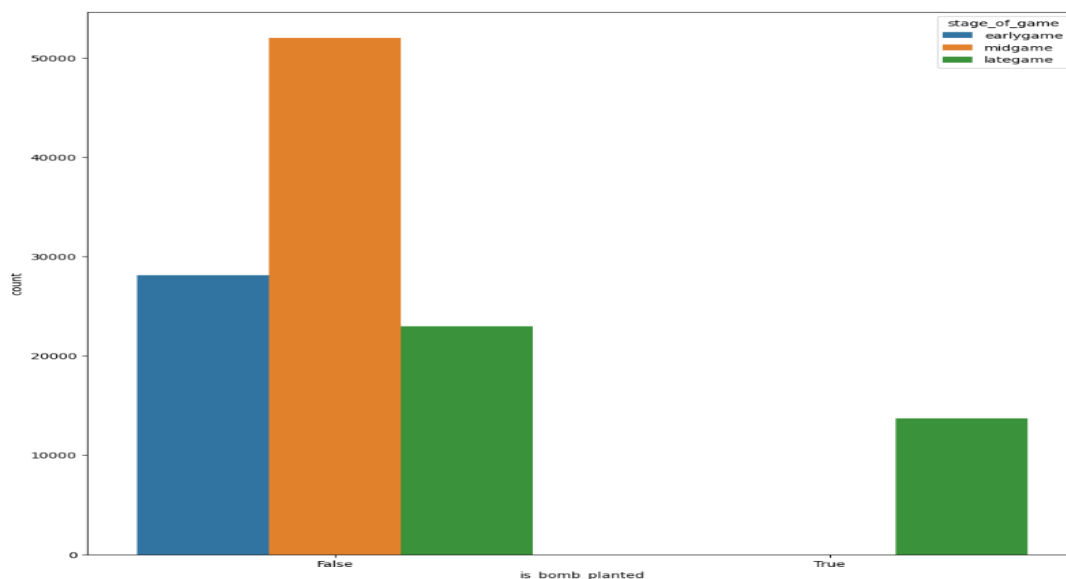


Рисунок 7 –фрагмент тепловой карты датасета

## 2.5 Диаграмма countplot с группировкой по двум номинальным признакам

CountPlot - столбчатая диаграмма, чаще всего используется для категориальных признаков в данных. Показывает, сколько строчек в датасете имеют каждое из выбранного значения категориального признака. Данная диаграмма показывает, что в earlygame и midgame бомба ни разу не была поставлена.





**Рисунок 8 – Диаграмма countplot по столбцам stage\_of\_game и is\_bomb\_planted**

### 3 Предварительная обработка данных

Данные имели пропущенные значения ,значит необходимо их заполнить. После заполнения получаем следующую картину:

```
In [56]: data.isnull().sum()
```

```
Out[56]: time_left      0
ct_score      0
t_score      0
map          0
is_bomb_planted 0
ct_health     0
t_health     0
ct_armor     0
t_armor     0
ct_money     0
t_money     0
ct_helmets   0
t_helmets   0
ct_defuse_kits 0
ct_players_alive 0
t_players_alive 0
stage_of_game 0
dtype: int64
```

**Рисунок 9 – Проверка на наличие пропусков в таблице**

Также было применено one-hot кодирование, то есть преобразование категориальных переменных в численные путем создания столбцов под каждую категорию и заполнения их значения 0 и 1.

In [58]:	data=pd.get_dummies(data) data												
Out[58]:	time_left	ct_score	t_score	is_bomb_planted	ct_health	t_health	ct_armor	t_armor	ct_money	t_money	...	map_de_dust2	
	0	175.00	0.0	0.0	False	500.0	500.0	0.0	0.0	4000.0	4000.0	...	1
	1	156.03	0.0	0.0	False	500.0	500.0	400.0	300.0	600.0	650.0	...	1
	2	96.03	0.0	0.0	False	391.0	400.0	294.0	200.0	750.0	500.0	...	1
	3	76.03	0.0	0.0	False	391.0	400.0	294.0	200.0	750.0	500.0	...	1
	4	174.97	1.0	0.0	False	500.0	500.0	192.0	0.0	18350.0	10750.0	...	1
	...	...	...	...	...	...	...	...	...	...	...	...	...
	122405	15.41	11.0	14.0	True	200.0	242.0	195.0	359.0	100.0	5950.0	...	0
	122406	174.93	11.0	15.0	False	500.0	500.0	95.0	175.0	11500.0	23900.0	...	0
	122407	114.93	11.0	15.0	False	500.0	500.0	495.0	475.0	1200.0	6700.0	...	0
	122408	94.93	11.0	15.0	False	500.0	500.0	495.0	475.0	1200.0	6700.0	...	0
	122409	74.93	11.0	15.0	False	375.0	479.0	395.0	466.0	1100.0	7000.0	...	0

116789 rows × 26 columns

**Рисунок 10 – Горячее кодирование**

Предобработанные данные были сохранены в формате .csv в той же директории, что и изначальный датасет.

9. Сохраните предобработанный набор данных в файл csv.

```
In [59]: data.to_csv('finally.csv')
```

**Рисунок 11 –Экспорт датасета**

## ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы, был проведен анализ курса по Python с библиотеками NumPy, Matplotlib, SymPy, SciPy и Seaborn, предоставляемого в университете. Этот курс охватывает широкий спектр тем, связанных с научным программированием и анализом данных, и предоставляет студентам ценные навыки для работы в современной сфере науки и технологий.

Основные выводы, сделанные в результате анализа курса:

1. Python как основной язык программирования обеспечивает простоту и удобство в изучении для студентов с различным уровнем подготовки. Богатство библиотек, таких как NumPy, Matplotlib, SymPy, SciPy и Seaborn, делает Python мощным инструментом для научных и инженерных расчетов, а также визуализации данных.
2. Библиотека NumPy предоставляет эффективные средства для работы с массивами и матрицами, что делает ее незаменимой для научных вычислений. Студенты могут легко выполнять операции линейной алгебры, статистические вычисления и многие другие задачи.
3. Matplotlib и Seaborn обеспечивают мощные средства для визуализации данных. С их помощью студенты могут создавать качественные графики, диаграммы и графики, что является важным навыком при анализе и представлении данных.
4. Библиотеки SymPy и SciPy позволяют студентам проводить символьные вычисления и решать сложные математические задачи. Эти библиотеки особенно полезны в инженерных и научных исследованиях.
5. Курс успешно сочетает в себе теоретические знания и практические навыки. Студенты имеют возможность применять полученные знания на практике, решая задачи и проекты, связанные с их дисциплиной.

В итоге, данный курс предоставляет студентам обширные знания и навыки, необходимые для эффективной работы в области научного программирования и анализа данных. Он подготавливает студентов к решению сложных задач и исследованиям. Эти навыки являются важными для программиста и обязательно потребуются.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

6. <https://numpy.org/doc/stable/reference/generated/numpy.matrix.html> (дата обращения: 23.10.23).
7. <https://seaborn.pydata.org/installing.html> (дата обращения: 21.10.23).
8. [https://pandas.pydata.org/docs/reference/api/pandas.get\\_dummies.html](https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html) (дата обращения: 20.10.23).
9. [https://matplotlib.org/stable/api/\\_as\\_gen/matplotlib.pyplot.tight\\_layout.html](https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.tight_layout.html) (дата обращения: 19.10.23).