Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Факультет «Информатика, искусственный и системы управления»

Кафедра «Системы обработки информации и управления»

Отчет по Лабораторной работе №2

«*Обработка пропусков в данных,*

кодирование категориальных признаков,

масштабирование данных.»

по дисциплине «Технология машинного обучения»

Выполнил:
студент группы ИУ5-61Б
И.А. Абуховский

Проверил:
Ю.Е. Гапанюк

2023 г.

Импорт

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import math as mth
import matplotlib.patches as patches
from scipy import stats as st
plt.rcParams.update({'figure.max_open_warning': 0})
import plotly.graph_objects as go
import plotly.express as px
```

In [2]:
```
df = pd.read_csv('season10.csv')
```

Обработка

In [3]:
```
df.head()
```

Out[3]:

	Game #	Start SR	End SR	SR Change	Team SR avg	Enemy SR avg	Team Stack	Enemy Stack	Role 1	Role 2	...	Obj_time_career	Obj_time_medal	Dmg	Dmg_career	Dn
0	1	P	P	NaN	P	P	2	2	Support	NaN	...	03:53	Gold	5074.0	6056.0	
1	2	P	P	NaN	P	P	4	5	Support	NaN	...	02:26	Gold	2257.0	4893.0	
2	3	P	P	NaN	P	P	3	3	Tank	NaN	...	01:48	None	7610.0	5414.0	
3	4	P	P	NaN	P	P	2	2	Tank	Offense	...	02:07	Gold	7458.0	5396.0	
4	5	P	P	NaN	P	P	3	3	Support	Tank	...	02:01	Bronze	2736.0	4890.0	

5 rows × 32 columns

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 99 entries, 0 to 98
Data columns (total 32 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Game #            99 non-null     int64  
 1   Start SR          99 non-null     object 
 2   End SR            99 non-null     object 
 3   SR Change         89 non-null     float64
 4   Team SR avg       99 non-null     object 
 5   Enemy SR avg      99 non-null     object 
 6   Team Stack        99 non-null     int64  
 7   Enemy Stack       99 non-null     int64  
 8   Role 1            99 non-null     object 
 9   Role 2            20 non-null     object 
 10  Result            99 non-null     object 
 11  Streak            99 non-null     int64  
 12  Leaver            99 non-null     object 
 13  Map               99 non-null     object 
 14  Match Time        98 non-null     object 
 15  Elim              98 non-null     float64
 16  Elim_career       98 non-null     float64
 17  Elim_medal        98 non-null     object 
 18  Obj_kills         98 non-null     float64
 19  Obj_kills_career  98 non-null     float64
 20  Obj_kills_medal   98 non-null     object 
 21  Obj_time          98 non-null     object 
 22  Obj_time_career   98 non-null     object 
 23  Obj_time_medal    98 non-null     object 
 24  Dmg               98 non-null     float64
 25  Dmg_career        98 non-null     float64
 26  Dmg_medal         98 non-null     object 
 27  Heal              98 non-null     float64
 28  Heal_career       98 non-null     float64
 29  Heal_medal        98 non-null     object 
 30  Death             98 non-null     float64
 31  Death_career      98 non-null     float64
dtypes: float64(11), int64(4), object(17)
memory usage: 24.9+ KB
```

Уберём пропуски

Явные пропуски

In [5]:
```
def draw_missing_data_table(df):
    total = df.isnull().sum().sort_values(ascending=False)
    percent = (df.isnull().sum()/df.isnull().count()).sort_values(ascending=False)*100
    missing_data = pd.concat([total, percent], axis=1, keys=['Total', 'Percent'])
    return missing_data
```

In [6]:
```
draw_missing_data_table(df)
```

Out[6]:

	Total	Percent
Role 2	79	79.797980
SR Change	10	10.101010
Elim_career	1	1.010101
Obj_kills_career	1	1.010101
Obj_kills_medal	1	1.010101
Obj_time	1	1.010101
Obj_time_career	1	1.010101
Obj_time_medal	1	1.010101
Dmg	1	1.010101
Dmg_career	1	1.010101
Elim_medal	1	1.010101
Dmg_medal	1	1.010101
Heal	1	1.010101
Heal_career	1	1.010101
Heal_medal	1	1.010101
Death	1	1.010101
Obj_kills	1	1.010101
Death_career	1	1.010101
Elim	1	1.010101
Match Time	1	1.010101
Start SR	0	0.000000
Map	0	0.000000
Leaver	0	0.000000
Streak	0	0.000000
Result	0	0.000000
Role 1	0	0.000000
Enemy Stack	0	0.000000
Team Stack	0	0.000000
Enemy SR avg	0	0.000000
Team SR avg	0	0.000000
End SR	0	0.000000
Game #	0	0.000000

In [15]:
```
df['Role 2'] = df['Role 2'].fillna('All Roles')
```

In [16]:
```
df.head()
```

Out[16]:

	Game #	Start SR	End SR	SR Change	Team SR avg	Enemy SR avg	Team Stack	Enemy Stack	Role 1	Role 2	...	Obj_time_career	Obj_time_medal	Dmg	Dmg_career	Dn
0	1	P	P	NaN	P	P	2	2	Support	All Roles	...	03:53	Gold	5074.0	6056.0	
1	2	P	P	NaN	P	P	4	5	Support	All Roles	...	02:26	Gold	2257.0	4893.0	
2	3	P	P	NaN	P	P	3	3	Tank	All Roles	...	01:48	None	7610.0	5414.0	
3	4	P	P	NaN	P	P	2	2	Tank	Offense	...	02:07	Gold	7458.0	5396.0	
4	5	P	P	NaN	P	P	3	3	Support	Tank	...	02:01	Bronze	2736.0	4890.0	

5 rows × 32 columns

In [17]:
```
draw_missing_data_table(df)
```

Out[17]:

	Total	Percent
SR Change	10	10.101010
Elim_career	1	1.010101
Elim_medal	1	1.010101
Death	1	1.010101
Heal_medal	1	1.010101
Heal_career	1	1.010101
Heal	1	1.010101
Dmg_medal	1	1.010101
Dmg_career	1	1.010101
Dmg	1	1.010101
Obj_time_medal	1	1.010101
Obj_time_career	1	1.010101
Obj_time	1	1.010101
Obj_kills_medal	1	1.010101
Obj_kills_career	1	1.010101
Obj_kills	1	1.010101
Death_career	1	1.010101
Elim	1	1.010101
Match Time	1	1.010101
Start SR	0	0.000000
Map	0	0.000000
Leaver	0	0.000000
Streak	0	0.000000
Result	0	0.000000
Role 2	0	0.000000
Role 1	0	0.000000
Enemy Stack	0	0.000000
Team Stack	0	0.000000
Enemy SR avg	0	0.000000
Team SR avg	0	0.000000
End SR	0	0.000000
Game #	0	0.000000

In [19]:
```
pd.get_dummies(df, columns=['Role 1']).head()
```

Out[19]:

	Game #	Start SR	End SR	SR Change	Team SR avg	Enemy SR avg	Team Stack	Enemy Stack	Role 2	Result	...	Dmg_medal	Heal	Heal_career	Heal_medal	Death	Deatl
0	1	P	P	NaN	P	P	2	2	All Roles	Win	...	None	8074.0	9636.0	Silver	6.0	
1	2	P	P	NaN	P	P	4	5	All Roles	Loss	...	Gold	4461.0	8367.0	Gold	8.0	
2	3	P	P	NaN	P	P	3	3	All Roles	Win	...	None	2132.0	5315.0	Bronze	10.0	
3	4	P	P	NaN	P	P	2	2	Offense	Loss	...	None	0.0	3533.0	None	16.0	
4	5	P	P	NaN	P	P	3	3	Tank	Win	...	None	3340.0	3503.0	Silver	5.0	

5 rows × 35 columns

Неявные пропуски

In [8]:
```
df.dropna(subset=['Game #'],inplace = True,axis = 0 )
```

In [9]:
```
df.describe().T
```

Out[9]:

	count	mean	std	min	25%	50%	75%	max
Game #	99.0	50.000000	28.722813	1.00	25.5000	50.000	74.50	99.00
SR Change	89.0	1.449438	23.740075	-30.00	-24.0000	19.000	23.00	35.00
Team Stack	99.0	2.010101	0.874759	1.00	1.0000	2.000	2.00	4.00
Enemy Stack	99.0	2.151515	0.993485	1.00	1.0000	2.000	3.00	5.00
Streak	99.0	0.505051	2.475899	-4.00	-1.0000	1.000	1.00	10.00
Elim	98.0	19.459184	9.967130	0.00	12.5000	19.000	25.00	56.00
Elim_career	98.0	15.494286	2.038293	0.00	15.1425	15.485	16.10	23.87
Obj_kills	98.0	9.846939	6.321869	0.00	5.0000	9.000	13.00	33.00
Obj_kills_career	98.0	8.404592	1.666424	0.00	8.0825	8.230	8.35	20.29
Dmg	98.0	5993.030612	3092.683409	0.00	3896.7500	5455.000	7594.75	13891.00
Dmg_career	98.0	4947.448980	789.886278	148.00	4768.7500	4955.500	5139.75	6387.00
Heal	98.0	5245.071429	4961.940559	0.00	0.0000	4824.500	9238.00	17258.00
Heal_career	98.0	3913.408163	2161.338073	0.00	3641.0000	4223.000	4390.00	9636.00
Death	98.0	9.051020	4.172498	0.00	6.0000	9.000	11.75	19.00
Death_career	98.0	7.832857	0.667582	3.95	7.4400	7.880	8.10	9.64

Проверим, какой процент пропусков будет составлять эти значения заменив их на NaN

In [10]:
```
data = df.copy()
data.replace(0, np.NaN,inplace=True)
```

In [12]:
```
draw_missing_data_table(data).round(1)
```

Out[12]:

	Total	Percent
Role 2	79	79.8
Heal	32	32.3
Heal_career	17	17.2
SR Change	13	13.1
Obj_kills	4	4.0
Streak	3	3.0
Elim	2	2.0
Dmg	2	2.0
Elim_career	2	2.0
Death	2	2.0
Obj_kills_career	2	2.0
Dmg_career	1	1.0
Dmg_medal	1	1.0
Heal_medal	1	1.0
Obj_time_medal	1	1.0
Obj_time_career	1	1.0
Obj_time	1	1.0
Obj_kills_medal	1	1.0
Death_career	1	1.0
Elim_medal	1	1.0
Match Time	1	1.0
Start SR	0	0.0
Map	0	0.0
Leaver	0	0.0
Result	0	0.0
Role 1	0	0.0
Enemy Stack	0	0.0
Team Stack	0	0.0
Enemy SR avg	0	0.0
Team SR avg	0	0.0
End SR	0	0.0
Game #	0	0.0

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js