Рубежный контроль №1 Вариант 9 Запруднов Михаил ИУ5-62б

Задание: Для заданного набора данных проведите обработку пропусков в данных для одного категориального и одного количественного признака. Какие способы обработки пропусков в данных для категориальных и количественных признаков Вы использовали? Какие признаки Вы будете использовать для дальнейшего построения моделей машинного обучения и почему?
Набор данных: https://www.kaggle.com/karangadiya/fifa19 + Построить гистограмму

B [60]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

**Загружаем данные:**

B [61]:
```
data = pd.read_csv('data.csv', sep=",")
```

B [62]:
```
# размер набора данных
data.shape
```

Out[62]:
```
(18207, 88)
```

B [63]:
```
# типы колонок
data.dtypes
```

Out[63]:
```
ID                 int64
Name              object
Age                int64
Photo             object
Nationality       object
                   ...
GKHandling       float64
GKKicking        float64
GKPositioning    float64
GKReflexes       float64
Release Clause    object
Length: 88, dtype: object
```

B [64]:
```
# определим пропуски в столбцах
data.isnull().sum()
```

Пропуски содержатся только в числовых данных

Out[64]:
```
ID                   0
Name                 0
Age                  0
Photo                0
Nationality          0
                   ...
GKHandling          48
GKKicking           48
GKPositioning       48
GKReflexes          48
Release Clause    1564
Length: 88, dtype: int64
```

B [65]:
```
# Первые 10 строк датасета
data.head(10)
```

Out[65]:

| | ID | Name | Age | Photo | Nationality | Flag | Overall | Potential | Club | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 158023 | L. Messi | 31 | https://cdn.sofifa.org/players/4/19/158023.png | Argentina | https://cdn.sofifa.org/flags/52.png | 94 | 94 | FC Barcelona | https://cdn.sofifa.org |
| 1 | 20801 | Cristiano Ronaldo | 33 | https://cdn.sofifa.org/players/4/19/20801.png | Portugal | https://cdn.sofifa.org/flags/38.png | 94 | 94 | Juventus | https://cdn.sofifa.or |
| 2 | 190871 | Neymar Jr | 26 | https://cdn.sofifa.org/players/4/19/190871.png | Brazil | https://cdn.sofifa.org/flags/54.png | 92 | 93 | Paris Saint-Germain | https://cdn.sofifa.or |
| 3 | 193080 | De Gea | 27 | https://cdn.sofifa.org/players/4/19/193080.png | Spain | https://cdn.sofifa.org/flags/45.png | 91 | 93 | Manchester United | https://cdn.sofifa.o |
| 4 | 192985 | K. De Bruyne | 27 | https://cdn.sofifa.org/players/4/19/192985.png | Belgium | https://cdn.sofifa.org/flags/7.png | 91 | 92 | Manchester City | https://cdn.sofifa.or |
| 5 | 183277 | E. Hazard | 27 | https://cdn.sofifa.org/players/4/19/183277.png | Belgium | https://cdn.sofifa.org/flags/7.png | 91 | 91 | Chelsea | https://cdn.sofifa.c |
| 6 | 177003 | L. Modrić | 32 | https://cdn.sofifa.org/players/4/19/177003.png | Croatia | https://cdn.sofifa.org/flags/10.png | 91 | 91 | Real Madrid | https://cdn.sofifa.org |
| 7 | 176580 | L. Suárez | 31 | https://cdn.sofifa.org/players/4/19/176580.png | Uruguay | https://cdn.sofifa.org/flags/60.png | 91 | 91 | FC Barcelona | https://cdn.sofifa.org |
| 8 | 155862 | Sergio Ramos | 32 | https://cdn.sofifa.org/players/4/19/155862.png | Spain | https://cdn.sofifa.org/flags/45.png | 91 | 91 | Real Madrid | https://cdn.sofifa.org |
| 9 | 200389 | J. Oblak | 25 | https://cdn.sofifa.org/players/4/19/200389.png | Slovenia | https://cdn.sofifa.org/flags/44.png | 90 | 93 | Atlético Madrid | https://cdn.sofifa.org |

10 rows × 88 columns

В [66]:
```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 18207
```

### Обработка пропуков в числовых данных

В [67]:
```
# Выберем числовые колонки с пропущенными значениями
# Цикл по колонкам датасета
num_cols = []
for col in data.columns:
    # Количество пустых значений
    temp_null_count = data[data[col].isnull()].shape[0]
    dt = str(data[col].dtype)
    if temp_null_count>0 and (dt=='float64' or dt=='int64'):
        num_cols.append(col)
        temp_perc = round((temp_null_count / total_count) * 100.0, 2)
        print('Колонка {}. Тип данных {}. Количество пустых значений {}, {}%.'.format(col, dt, temp_null_count, temp_perc))
```

```
Колонка International Reputation. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Weak Foot. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Skill Moves. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Jersey Number. Тип данных float64. Количество пустых значений 60, 0.33%.
Колонка Crossing. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Finishing. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка HeadingAccuracy. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка ShortPassing. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Volleys. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Dribbling. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Curve. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка FKAccuracy. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка LongPassing. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка BallControl. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Acceleration. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка SprintSpeed. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Agility. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Reactions. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Balance. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка ShotPower. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Jumping. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Stamina. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Strength. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка LongShots. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Aggression. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Interceptions. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Positioning. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Vision. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Penalties. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Composure. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка Marking. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка StandingTackle. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка SlidingTackle. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка GKDiving. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка GKHandling. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка GKKicking. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка GKPositioning. Тип данных float64. Количество пустых значений 48, 0.26%.
Колонка GKReflexes. Тип данных float64. Количество пустых значений 48, 0.26%.
```

**В колонках с отсутствующими значениями содержится информация о характеристиках футболистов, т.к. мы не можем узнать значения их характеристик, заполним их значениями, средними по данному столбцу.**

В [72]:
```
# Выберем числовые колонки с пропущенными значениями
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений
    temp_null_count = data[data[col].isnull()].shape[0]
    dt = str(data[col].dtype)
    if temp_null_count>0 and (dt=='float64' or dt=='int64'):
        data[col] = data[col].fillna(round(data[[col]].mean()[0], 3))
    if temp_null_count>0 and (dt=='object'):
        data[col] = data[col].fillna('€0M')
```

В столбце Release Clause указаны значения в евро.

```
В [73]: data[['Release Clause']]
```

Out[73]:

| | Release Clause |
|---|---|
| 0 | €226.5M |
| 1 | €127.1M |
| 2 | €228.1M |
| 3 | €138.6M |
| 4 | €196.4M |
| ... | ... |
| 18202 | €143K |
| 18203 | €113K |
| 18204 | €165K |
| 18205 | €143K |
| 18206 | €165K |

18207 rows × 1 columns

```
В [74]: data.head()
```

Out[74]:

| | ID | Name | Age | Photo | Nationality | Flag | Overall | Potential | Club | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 158023 | L. Messi | 31 | https://cdn.sofifa.org/players/4/19/158023.png | Argentina | https://cdn.sofifa.org/flags/52.png | 94 | 94 | FC Barcelona | https://cdn.sofifa.org |
| 1 | 20801 | Cristiano Ronaldo | 33 | https://cdn.sofifa.org/players/4/19/20801.png | Portugal | https://cdn.sofifa.org/flags/38.png | 94 | 94 | Juventus | https://cdn.sofifa.o |
| 2 | 190871 | Neymar Jr | 26 | https://cdn.sofifa.org/players/4/19/190871.png | Brazil | https://cdn.sofifa.org/flags/54.png | 92 | 93 | Paris Saint-Germain | https://cdn.sofifa.o |
| 3 | 193080 | De Gea | 27 | https://cdn.sofifa.org/players/4/19/193080.png | Spain | https://cdn.sofifa.org/flags/45.png | 91 | 93 | Manchester United | https://cdn.sofifa.o |
| 4 | 192985 | K. De Bruyne | 27 | https://cdn.sofifa.org/players/4/19/192985.png | Belgium | https://cdn.sofifa.org/flags/7.png | 91 | 92 | Manchester City | https://cdn.sofifa.o |

5 rows × 88 columns

```
В [75]: # проверим пропуски в столбцах
data.isnull().sum()
```

Out[75]:

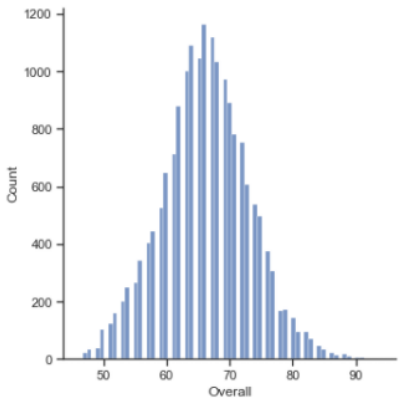
```
ID                0
Name              0
Age               0
Photo             0
Nationality       0
                 ..
GKHandling        0
GKKicking         0
GKPositioning     0
GKReflexes        0
Release Clause    0
Length: 88, dtype: int64
```

**Построим гистограмму для столбца Overall**

```
В [59]: sns.displot(x=data['Overall'])
```

Out[59]:

```
<seaborn.axisgrid.FacetGrid at 0x1b1bb8e82e0>
```




**Выводы, ответы на вопросы к РК:** В данной работе для обработки пропусков данных мы воспользовались двумя стратегиями: 1) заполнением нулями признака у которого нет возможности узнать средние значение, или точное значение 2) импутация данных в признаке, в котором количество пропусков не превышает порогового значения (5%), путем заполнения наиболее средним значением (вывод о применимости моды был сделан исходя из гистограммы

распределения). Кроме того, исследование количества пропусков дают одинаковые показатели, а значит, скорее всего, эти пропуски сделаны в одних строках. В дальнейшем можно просто удалить данный строки, либо дополнить их верными данными. Пропусков в категориальных признаках обнаружено не было. Окончательное решение по выбору признаков, поступающих на вход модели, может приниматься после проведения корреляционного анализа. Также после проведения кросс-валидации и подбора оптимальных параметров модели возможен пересмотр набора признаков: либо их удаление, либо их добавление в зависимости от результатов работы алгоритма машинного обучения.