Министерство образования и науки 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)

Отчёт по рубежному контролю №1 по курсу «Методы машинного обучения»

Вариант 8/28

Выполнил: Пименов Г.Ю.

Группа: ИУ5-24М

Москва, 2023

Задание на РК1:

- Для набора данных проведите устранение пропусков для одного (произвольного) числового признака с использованием метода заполнения модой
- Для набора данных для одного (произвольного) числового признака проведите обнаружение и замену (найденными верхними и нижними границами) выбросов на основе межквартильного размаха.
- Для произвольной колонки данных построить график “Скрипичная диаграмма”

Выполнение:

Загружаем необходимое для работы:

In [16]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import scipy.stats as stats
from sklearn.impute import SimpleImputer
from sklearn.impute import MissingIndicator
from sklearn.impute import KNNImputer
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Lasso
from sklearn.pipeline import Pipeline
from sklearn.model_selection import GridSearchCV
from sklearn.ensemble import RandomForestRegressor
from sklearn.experimental import enable_iterative_imputer
from sklearn.impute import IterativeImputer
%matplotlib inline
sns.set(style="ticks")
```

Загружаем датасет и проверяем:

In [2]:
```
data = pd.read_csv("gamesales.csv")
```

In [3]:
```
data.head(10)
```

Out[3]:

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo	41.49	29.02	3.77
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58	6.81
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.85	12.88	3.79
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.75	11.01	3.28
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89	10.22
5	6	Tetris	GB	1989.0	Puzzle	Nintendo	23.20	2.26	4.22
6	7	New Super Mario Bros.	DS	2006.0	Platform	Nintendo	11.38	9.23	6.50
7	8	Wii Play	Wii	2006.0	Misc	Nintendo	14.03	9.20	2.93
8	9	New Super Mario Bros. Wii	Wii	2009.0	Platform	Nintendo	14.59	7.06	4.70
9	10	Duck Hunt	NES	1984.0	Shooter	Nintendo	26.93	0.63	0.28

Задание 1

Определяем пропущенные значения в столбцах:

```
In [4]: data.shape
Out[4]: (16598, 11)

In [5]: data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Rank          16598 non-null  int64
 1   Name          16598 non-null  object
 2   Platform      16598 non-null  object
 3   Year          16327 non-null  float64
 4   Genre         16598 non-null  object
 5   Publisher     16540 non-null  object
 6   NA_Sales      16598 non-null  float64
 7   EU_Sales      16598 non-null  float64
 8   JP_Sales      16598 non-null  float64
 9   Other_Sales   16598 non-null  float64
 10  Global_Sales  16598 non-null  float64
dtypes: float64(6), int64(1), object(4)
memory usage: 1.4+ MB

In [7]: data.isna().sum()
Out[7]: Rank            0
        Name            0
        Platform        0
        Year          271
        Genre           0
        Publisher      58
        NA_Sales        0
        EU_Sales        0
        JP_Sales        0
        Other_Sales     0
        Global_Sales    0
        dtype: int64
```

Решаем задачу:

In [11]:
```
# Пропущенные значения в стобцах Year и Publisher
# Задача 8:
# Для набора данных проведите устранение пропусков для одного (произвольного)
# с использованием метода заполнения модой

mode_value = data['Year'].mode()[0]

# Нашли наиболее часто встречающееся или "модное" значение в столбце

data['Year'].fillna(mode_value, inplace=True)

# Заменили пропуски на это значение

data.isna().sum()
```

Out[11]:
```
Rank            0
Name            0
Platform        0
Year            0
Genre           0
Publisher      58
NA_Sales        0
EU_Sales        0
JP_Sales        0
Other_Sales     0
Global_Sales    0
dtype: int64
```

In [12]:
```
# Как видим, пропуски в столбце Year пропали
```

```
# Повторим алгоритм для столбца Publisher

mode_value = data['Publisher'].mode()[0]
data['Publisher'].fillna(mode_value, inplace=True)
data.isna().sum()
```

Out[12]:
```
Rank            0
Name            0
Platform        0
Year            0
Genre           0
Publisher       0
NA_Sales        0
EU_Sales        0
JP_Sales        0
Other_Sales     0
Global_Sales    0
dtype: int64
```

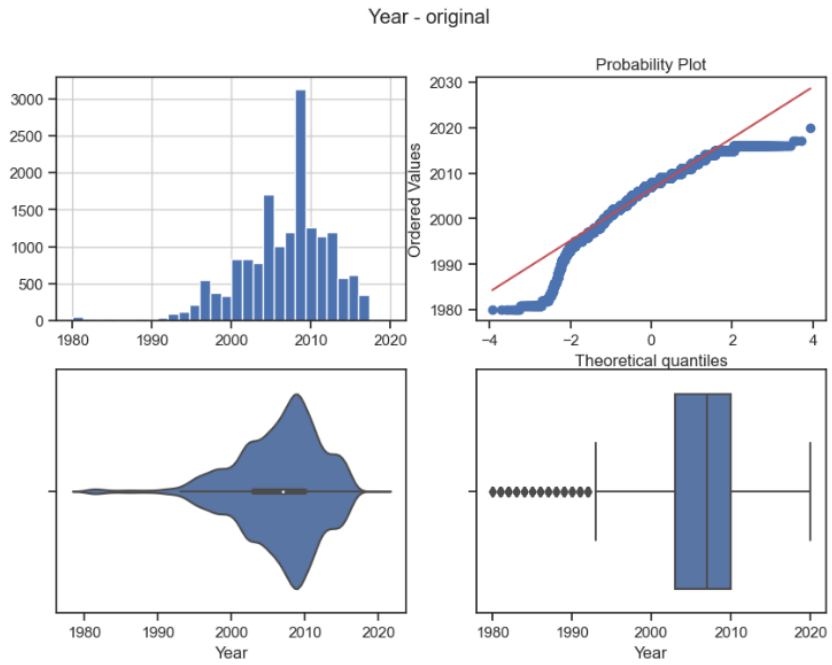
Задание 2

Решаем задачу:

In [14]:
```
# Задача 28:
# Для набора данных для одного (произвольного) числового пр
# (найденными верхними и нижними границами) выбросов на осн

def diagnostic_plots(df, variable, title):
    fig, ax = plt.subplots(figsize=(10,7))
    # гистограмма
    plt.subplot(2, 2, 1)
    df[variable].hist(bins=30)
    ## Q-Q plot
    plt.subplot(2, 2, 2)
    stats.probplot(df[variable], dist="norm", plot=plt)
    # ящик с усами
    plt.subplot(2, 2, 3)
    sns.violinplot(x=df[variable])
    # ящик с усами
    plt.subplot(2, 2, 4)
    sns.boxplot(x=df[variable])
    fig.suptitle(title)
    plt.show()
```

In [17]:
```
diagnostic_plots(data, 'Year', 'Year - original')
```



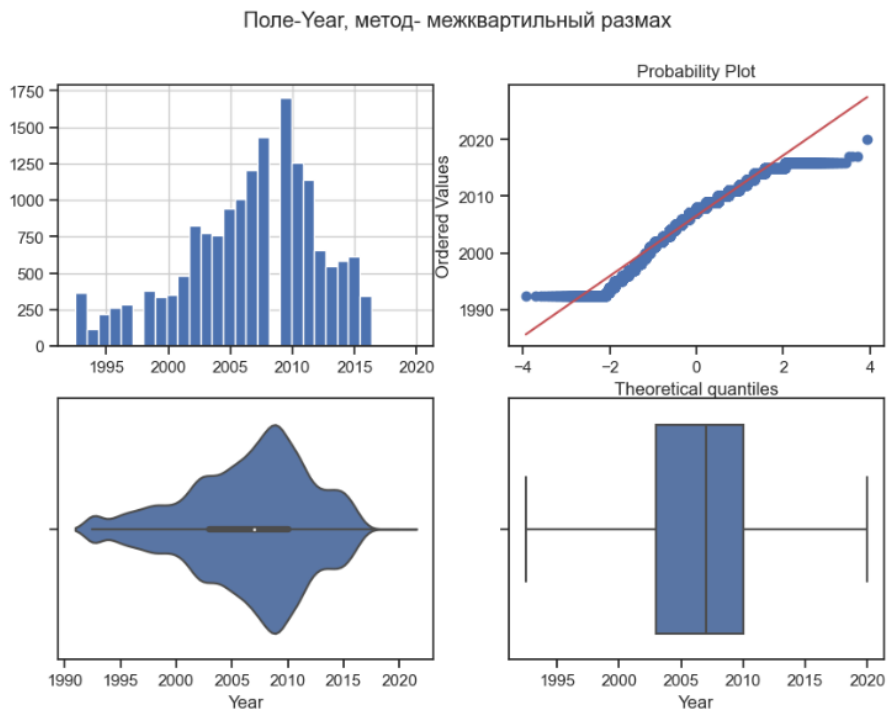
In [23]:
```
# Функция вычисления верхней и нижней границы выбросов

def get_outlier_boundaries(df, col):

  K2 = 1.5
  IQR = df[col].quantile(0.75) - df[col].quantile(0.25)
  lower_boundary = df[col].quantile(0.25) - (K2 * IQR)
  upper_boundary = df[col].quantile(0.75) + (K2 * IQR)


  return lower_boundary, upper_boundary
```

In [25]:
```
    col = 'Year'
    # Вычисление верхней и нижней границы
    lower_boundary, upper_boundary = get_outlier_boundaries(data, col)
    # Изменение данных
    data[col] = np.where(data[col] > upper_boundary, upper_boundary,
                         np.where(data[col] < lower_boundary, lower_bound
    title = 'Поле-{}, метод- межквартильный размах'.format(col)
    diagnostic_plots(data, col, title)
```



Дополнительное задание:

In [26]:
```
# Дополнительное задание: для произвольной колонки данных построить график "
sns.violinplot(x='Year', data=data)
```

Out[26]: <AxesSubplot:xlabel='Year'>