**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 16/36**

Выполнил: Ульбашев А.Н

Группа: ИУ5-22М

Москва, 2023

## Задание на РК:

- Для набора данных проведите нормализацию для одного (произвольного) числового признака с использованием преобразования Бокса-Кокса (Box-Cox transformation).
- Для набора данных проведите процедуру отбора признаков (feature selection). Используйте класс SelectKBest для 5 лучших признаков, и метод, основанный на взаимной информации.
- Для произвольной колонки данных построить гистограмму.

## Выполнение:

Загружаем необходимые для работы инструменты:

```
!pip install pandas
!pip install seaborn
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Загружаем датасет и проверяем готовность его к работе:

```
df = pd.read_csv('/content/movies.csv')
```

---

**Избавляемся от пропусков путем удаления строк с пропусками**

---

```
df.isna().mean()
```

```
Title               0.000
Rating              0.001
Year                0.000
Month               0.000
Certificate         0.017
Runtime             0.000
Directors           0.000
Stars               0.000
Genre               0.000
Filming_location    0.000
Budget              0.000
Income              0.000
Country_of_origin   0.000
dtype: float64
```

Готовим датасет к работе:

```
df = df.dropna(axis=0, how='any')
```

```
df.isna().mean()
```

```
Title              0.0
Rating             0.0
Year               0.0
Month              0.0
Certificate        0.0
Runtime            0.0
Directors          0.0
Stars              0.0
Genre              0.0
Filming_location   0.0
Budget             0.0
Income             0.0
Country_of_origin  0.0
dtype: float64
```

Задание 1:

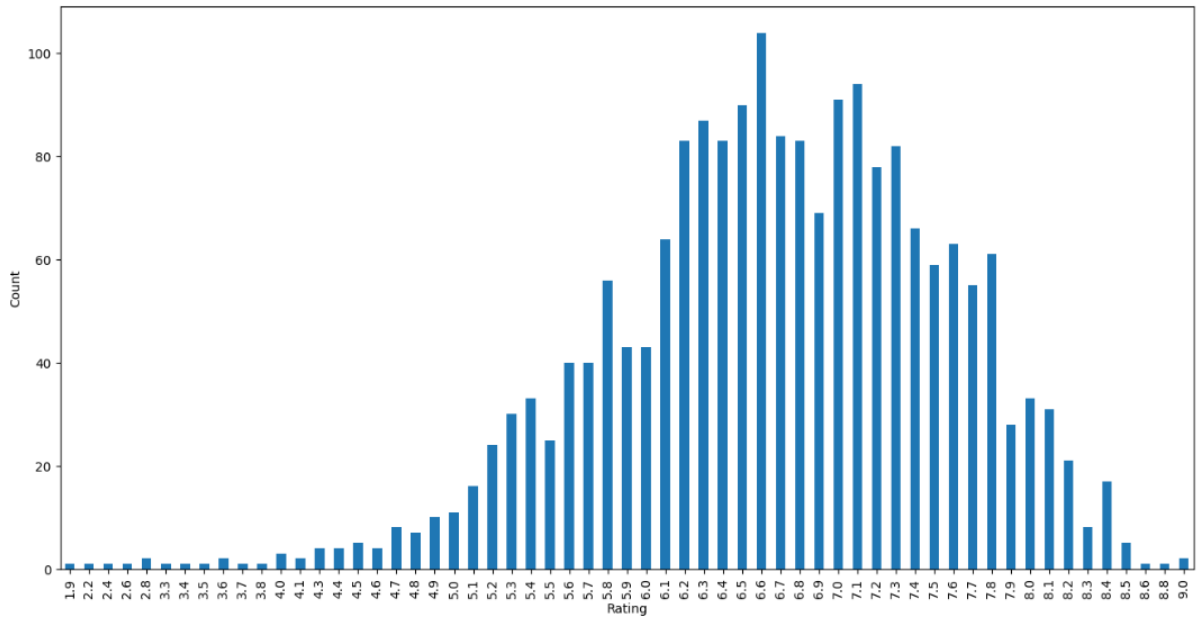
---

**Для набора данных проведите нормализацию для одного (произвольного) числового признака с использованием преобразования Бокса-Кокса (Box-Cox transformation).**

---

```
[9]: df.groupby('Rating')['Year'].agg('count').plot(kind = 'bar',figsize=( 16 , 8 ), ylabel = 'Count')
```

```
[9]: <Axes: xlabel='Rating', ylabel='Count'>
```

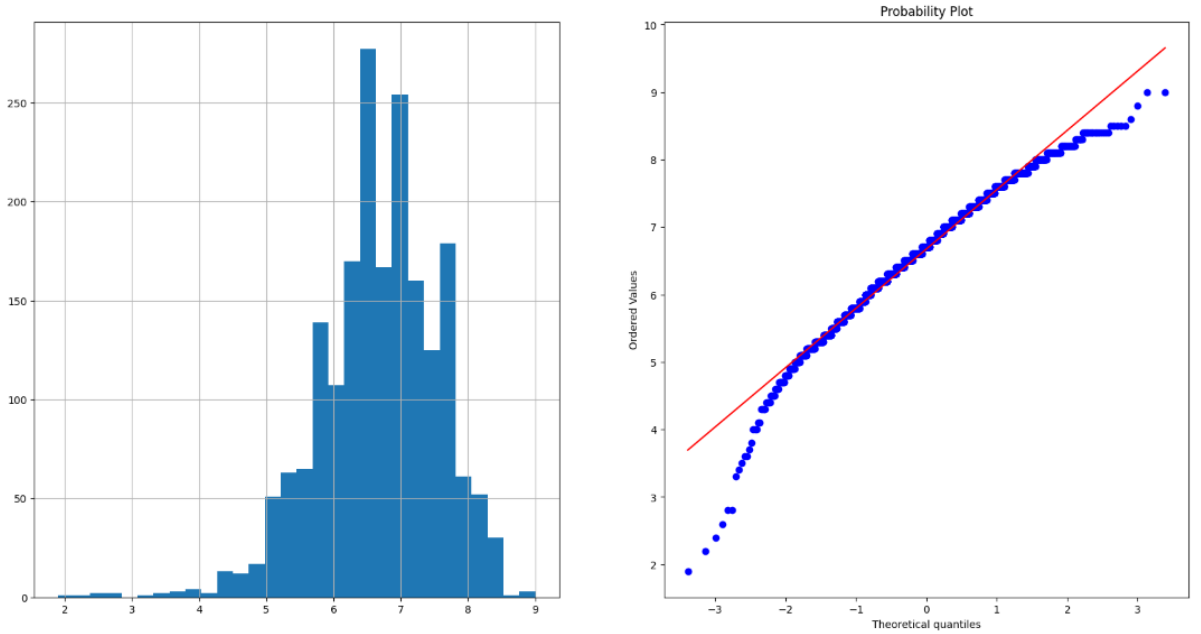


In [7]:
```
import scipy.stats as stats
def diagnostic_plots(df, variable):
    plt.figure(figsize=(20,10))
    # гистограмма
    plt.subplot(1, 2, 1)
    df[variable].hist(bins=30)
    ## Q-Q plot
    plt.subplot(1, 2, 2)
    stats.probplot(df[variable], dist="norm", plot=plt)
    plt.show()
```

In [8]:
```
df['Rating_boxcox'], param = stats.boxcox(df['Rating'])
print('Оптимальное значение λ = {}'.format(param))
diagnostic_plots(df, 'Rating')
```

```
Оптимальное значение λ = 2.2399056678483884
```

Оптимальное значение λ = 2.2399056678483884




## Задание 2:

Загружаем датасет отражающий некоторые показатели здоровья и наличие сахарного диабета и разбиваем на целевой массив и массив данных.

**Для набора данных проведите процедуру отбора признаков (feature selection). Используйте класс SelectKBest для 5 лучших признаков, и метод, основанный на взаимной информации.**

---

```
from sklearn.feature_selection import mutual_info_classif, mutual_info_regression, f_regression
from sklearn.feature_selection import SelectKBest, SelectPercentile
```

```
df2= pd.read_csv('/content/diabetes.csv')
```

```
dfX=df2[['Pregnancies','BloodPressure','SkinThickness','BMI','DiabetesPedigreeFunction','Glucose', 'Insulin', 'Age']]
dfY=df2[['Outcome']]
df3=df2.drop(columns= 'Outcome')
df2_feature_names= list(df3.columns)
```

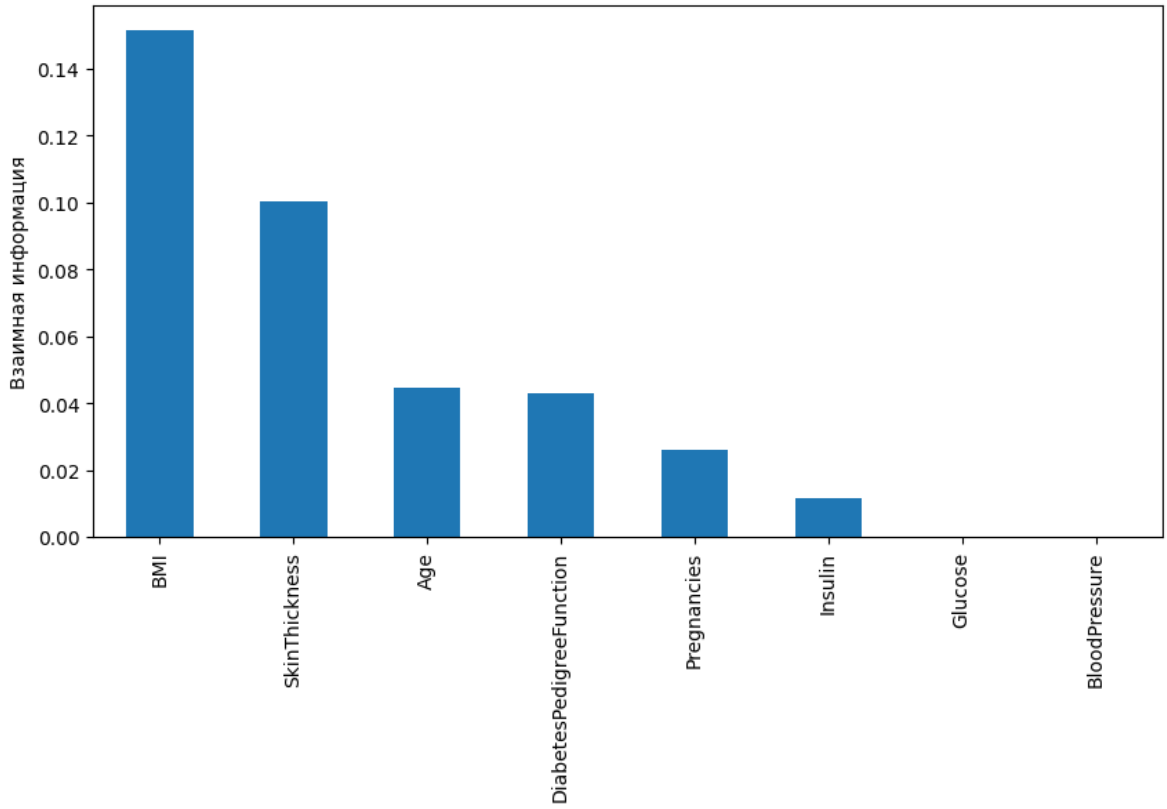
```
mi = mutual_info_regression(dfX, dfY)
mi = pd.Series(mi)
mi.index = df2_feature_names
mi.sort_values(ascending=False).plot.bar(figsize=(10,5))
plt.ylabel('Взаимная информация')
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/utils/validation.py:1143: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Text(0, 0.5, 'Взаимная информация')
```

Строим гистограмму.

```
: Text(0, 0.5, 'Взаимная информация')
```




Выбираем 5 лучших:

```
In [65]: sel_mi = SelectKBest(mutual_info_regression, k=5).fit(dfX, dfY)
         list(zip(df2, sel_mi.get_support()))
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/utils/validation.py:1143: DataConversionWarning: A column-vector y was passed when a 1d array was exp
ected. Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
Out[65]: [('Pregnancies', False),
          ('Glucose', False),
          ('BloodPressure', True),
          ('SkinThickness', True),
          ('Insulin', False),
          ('BMI', True),
          ('DiabetesPedigreeFunction', True),
          ('Age', True)]
```