***Рубежный контроль №1*  
Студент: Уляшин В.В.  
Группа: ИУ5-23М**

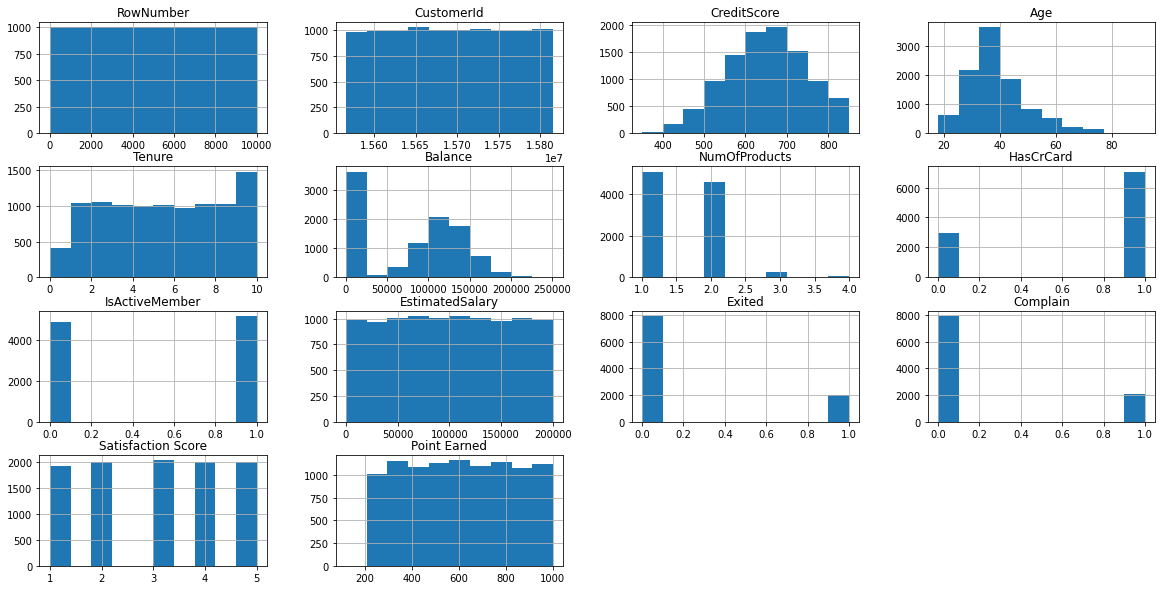
import pandas as pd  
import matplotlib.pyplot as plt  
import scipy.stats as stats  
from sklearn.datasets import load\_wine  
from sklearn.neighbors import KNeighborsClassifier   
from mlxtend.feature\_selection import ExhaustiveFeatureSelector as EFS   
import seaborn as sns

Задача №13. Для набора данных проведите нормализацию для одного (произвольного) числового признака с использованием функции "обратная зависимость - 1 / X".

data = pd.read\_csv("Customer-Churn-Records.csv")  
data.shape

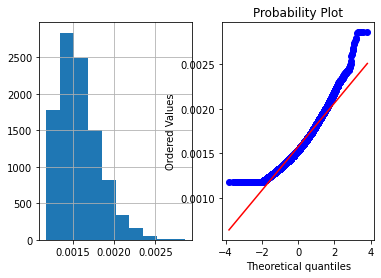
(10000, 18)

data.hist(figsize=(20,10))  
plt.show()



data['normalized\_feature'] = data['CreditScore'].apply(lambda x: 1 / x)

plt.subplot(1,2,1)  
data['normalized\_feature'].hist()  
plt.subplot(1,2,2)  
stats.probplot(data['normalized\_feature'], dist="norm", plot=plt)  
plt.show()



Задача №33. Для набора данных проведите процедуру отбора признаков (feature selection). Используйте метод обертывания (wrapper method), алгоритм полного перебора (exhaustive feature selection).

wine = load\_wine()  
data = pd.DataFrame(data=wine.data, columns= wine.feature\_names)  
y = wine['target']

knn = KNeighborsClassifier(n\_neighbors=3)  
efs1 = EFS(knn,  
 min\_features=2,  
 max\_features=4,  
 scoring='accuracy',  
 print\_progress=True,  
 cv=5)  
efs1 = efs1.fit(data, y)

Features: 1079/1079

print('Best accuracy score: %.2f' % efs1.best\_score\_)  
print('Best subset:', efs1.best\_feature\_names\_)

Best accuracy score: 0.94  
Best subset: ('alcohol', 'total\_phenols', 'flavanoids', 'color\_intensity')

knn = KNeighborsClassifier(n\_neighbors=3)  
efs1 = EFS(knn,  
 min\_features=1,  
 max\_features=2,  
 scoring='accuracy',  
 print\_progress=True,  
 cv=5)  
efs1 = efs1.fit(data, y)

Features: 91/91

print('Best accuracy score: %.2f' % efs1.best\_score\_)  
print('Best subset:', efs1.best\_feature\_names\_)

Best accuracy score: 0.93  
Best subset: ('flavanoids', 'color\_intensity')

Для студентов групп ИУ5-23М, ИУ5И-23М - для произвольной колонки данных построить график "Ящик с усами (boxplot)".

data = load\_wine()  
df = pd.DataFrame(data=data.data, columns=wine.feature\_names)  
df['target'] = wine.target

sns.boxplot(data=df, x='target', y='alcohol')  
plt.suptitle('Взаимосвязь между призанками и метками классов')  
plt.show()

