# Рубежный контроль №1

## Усков Д.Ю. Группа ИУ5-63Б

### Вариант 22

**Задача**. Для заданного набора данных произведите масштабирование данных (для одного признака) и преобразование категориальных признаков в количественные двумя способами (label encoding, one hot encoding) для одного признака. Какие методы Вы использовали для решения задачи и почему?

**Дополнительное требование:** для произвольной колонки данных построить график "Ящик с усами (boxplot)".

**Набор данных:** https://www.kaggle.com/rhuebner/human-resources-data-set

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
# загрузка набора данных
data = pd.read_csv('HRDataset_v14.csv', sep=",")
# размер набора данных
data.shape
```

Out[2]:
```
(311, 36)
```

In [3]:
```
# первые 5 строк набора данных
data.head()
```

Out[3]:

| | Employee_Name | EmpID | MarriedID | MaritalStatusID | GenderID | EmpStatusID | DeptID | PerfScoreID |
|---|---|---|---|---|---|---|---|---|
| **0** | Adinolfi, Wilson K | 10026 | 0 | 0 | 1 | 1 | 5 | 4 |
| **1** | Ait Sidi, Karthikeyan | 10084 | 1 | 1 | 1 | 5 | 3 | 3 |
| **2** | Akinkuolie, Sarah | 10196 | 1 | 1 | 0 | 5 | 5 | 3 |
| **3** | Alagbe,Trina | 10088 | 1 | 1 | 0 | 1 | 5 | 3 |
| **4** | Anderson, Carol | 10069 | 0 | 2 | 0 | 5 | 5 | 3 |

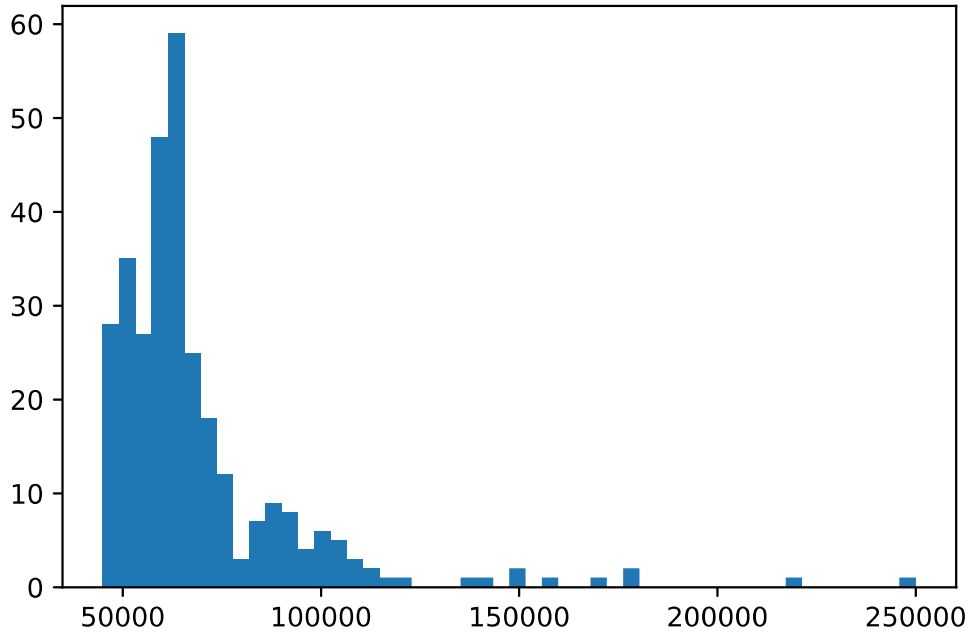
5 rows × 36 columns

# Масштабирование данных:

Для решения этой задачи я буду использовать MinMax масштабирование.

Например, произведем масштабирование признака "Salary":

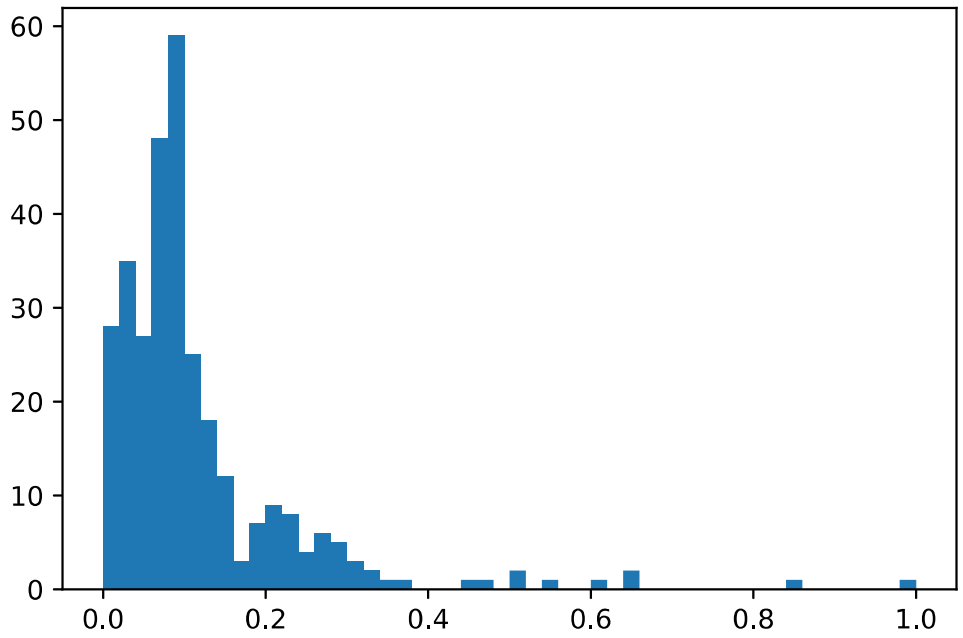
In [4]:
```
# гистограмма распределения данного признака
plt.hist(data['Salary'], 50)
plt.show()
```




In [5]:
```
from sklearn.preprocessing import MinMaxScaler
```

In [6]:
```
# MinMax масштабирование
mms = MinMaxScaler()
sc_data = mms.fit_transform(data[['Salary']])
```

In [7]:
```
# гистограмма распределения после MinMax масгтабирования данного признака
plt.hist(sc_data, 50)
plt.show()
```



# Преобразование категориальных признаков в количественные:

## One-hot encoding:

Например, выполним преобразование для категориального признака "RecruitmentSource":

In [8]:
```
# one-hot encoding
pd.get_dummies(data['RecruitmentSource']).head()
```

Out[8]:

| | CareerBuilder | Diversity Job Fair | Employee Referral | Google Search | Indeed | LinkedIn | On-line Web application | Other | Website |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **1** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **2** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **3** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **4** | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

## Label encoding:

In [9]:
```
from sklearn.preprocessing import LabelEncoder
```

In [10]:
```
# исходные уникальные значения данного признака
data['RecruitmentSource'].unique()
```

Out[10]:
```
array(['LinkedIn', 'Indeed', 'Google Search', 'Employee Referral',
       'Diversity Job Fair', 'On-line Web application', 'CareerBuilder',
       'Website', 'Other'], dtype=object)
```

In [11]:
```
# label encoding
le = LabelEncoder()
data_le = le.fit_transform(data['RecruitmentSource'])
```

In [12]:
```
# уникальные значения после label encoding
np.unique(data_le)
```

Out[12]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8])
```

In [13]:
```
# обратное преобразование
le.inverse_transform(data_le)
```

Out[13]:
```
array(['LinkedIn', 'Indeed', 'LinkedIn', 'Indeed', 'Google Search',
       'LinkedIn', 'LinkedIn', 'Employee Referral', 'Diversity Job Fair',
       'Indeed', 'Diversity Job Fair', 'Diversity Job Fair',
       'Diversity Job Fair', 'Google Search', 'On-line Web application',
       'Google Search', 'Employee Referral', 'Google Search',
       'Google Search', 'LinkedIn', 'Google Search', 'Indeed', 'Indeed',
       'CareerBuilder', 'Google Search', 'LinkedIn', 'Diversity Job Fair',
       'Indeed', 'Google Search', 'Diversity Job Fair', 'Google Search',
       'Diversity Job Fair', 'Google Search', 'Employee Referral',
       'Indeed', 'Google Search', 'Indeed', 'Indeed', 'LinkedIn',
       'LinkedIn', 'Indeed', 'Google Search', 'Indeed', 'Indeed',
```

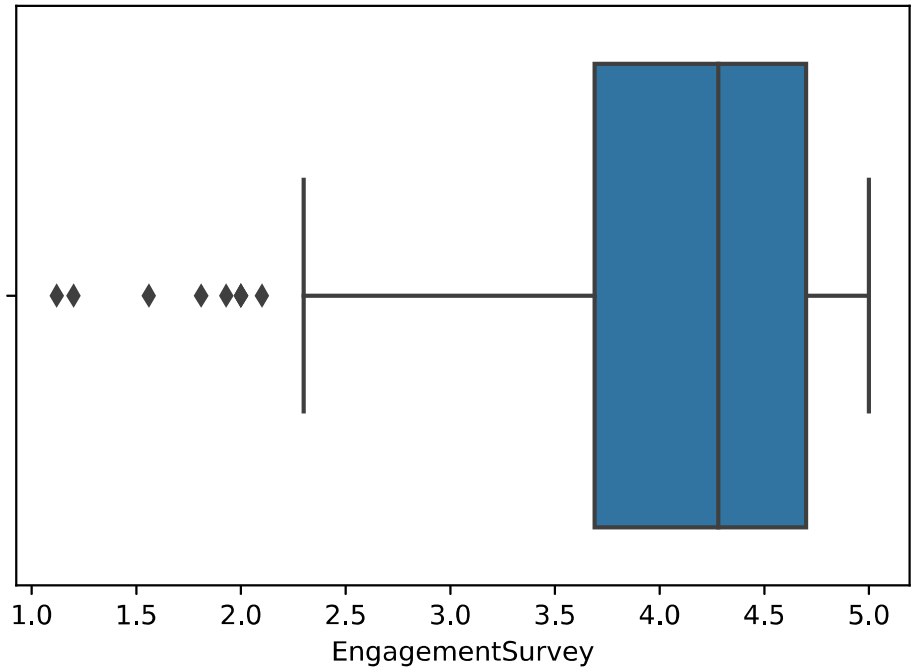
```
'LinkedIn', 'Employee Referral', 'Indeed', 'Indeed', 'Indeed',
'Google Search', 'Indeed', 'Employee Referral',
'Employee Referral', 'LinkedIn', 'CareerBuilder', 'Indeed',
'Indeed', 'Indeed', 'LinkedIn', 'Employee Referral', 'Indeed',
'LinkedIn', 'Indeed', 'LinkedIn', 'Indeed', 'Indeed',
'CareerBuilder', 'Indeed', 'Google Search', 'Indeed', 'Indeed',
'Indeed', 'Website', 'Indeed', 'CareerBuilder',
'Employee Referral', 'Indeed', 'Indeed', 'Google Search',
'Google Search', 'Google Search', 'LinkedIn', 'Employee Referral',
'Indeed', 'Google Search', 'Google Search', 'Indeed', 'LinkedIn',
'LinkedIn', 'Google Search', 'Indeed', 'LinkedIn', 'Google Search',
'Google Search', 'Google Search', 'Employee Referral', 'Indeed',
'Other', 'CareerBuilder', 'LinkedIn', 'Employee Referral',
'LinkedIn', 'LinkedIn', 'Diversity Job Fair', 'CareerBuilder',
'Diversity Job Fair', 'Indeed', 'Indeed', 'Indeed', 'LinkedIn',
'Indeed', 'Diversity Job Fair', 'Diversity Job Fair',
'Employee Referral', 'LinkedIn', 'LinkedIn', 'Google Search',
'LinkedIn', 'Employee Referral', 'CareerBuilder', 'Indeed',
'LinkedIn', 'Google Search', 'LinkedIn', 'CareerBuilder',
'CareerBuilder', 'Google Search', 'CareerBuilder', 'Indeed',
'Indeed', 'Indeed', 'LinkedIn', 'Indeed', 'LinkedIn', 'LinkedIn',
'Indeed', 'Google Search', 'Indeed', 'Indeed', 'LinkedIn',
'Employee Referral', 'LinkedIn', 'LinkedIn', 'CareerBuilder',
'Indeed', 'LinkedIn', 'CareerBuilder', 'Google Search', 'Indeed',
'Indeed', 'Indeed', 'Google Search', 'Google Search', 'LinkedIn',
'Indeed', 'Diversity Job Fair', 'Employee Referral',
'Employee Referral', 'Indeed', 'LinkedIn', 'Website',
'Google Search', 'Indeed', 'CareerBuilder', 'CareerBuilder',
'Google Search', 'Website', 'Website', 'Indeed', 'LinkedIn',
'Diversity Job Fair', 'LinkedIn', 'Indeed', 'Website',
'Diversity Job Fair', 'Indeed', 'LinkedIn', 'LinkedIn', 'LinkedIn',
'LinkedIn', 'Google Search', 'Indeed', 'LinkedIn', 'CareerBuilder',
'Website', 'Diversity Job Fair', 'CareerBuilder', 'Indeed',
'LinkedIn', 'LinkedIn', 'Diversity Job Fair', 'Indeed',
'Diversity Job Fair', 'CareerBuilder', 'LinkedIn', 'LinkedIn',
'LinkedIn', 'Indeed', 'Diversity Job Fair', 'Employee Referral',
'LinkedIn', 'Diversity Job Fair', 'Indeed', 'LinkedIn', 'Indeed',
'LinkedIn', 'LinkedIn', 'Indeed', 'LinkedIn', 'LinkedIn',
'Website', 'Google Search', 'Diversity Job Fair',
'Employee Referral', 'Google Search', 'CareerBuilder', 'LinkedIn',
'Indeed', 'Indeed', 'LinkedIn', 'Employee Referral',
'Google Search', 'Google Search', 'Website', 'Google Search',
'Employee Referral', 'Indeed', 'Diversity Job Fair', 'Indeed',
'Indeed', 'CareerBuilder', 'LinkedIn', 'Google Search',
'Google Search', 'Indeed', 'Indeed', 'Indeed', 'Employee Referral',
'Employee Referral', 'LinkedIn', 'Indeed', 'Website',
'Google Search', 'Indeed', 'Diversity Job Fair', 'Indeed',
'Diversity Job Fair', 'Google Search', 'CareerBuilder', 'LinkedIn',
'LinkedIn', 'Google Search', 'LinkedIn', 'LinkedIn',
'Employee Referral', 'Website', 'CareerBuilder', 'Indeed',
'Diversity Job Fair', 'Diversity Job Fair', 'Employee Referral',
'LinkedIn', 'LinkedIn', 'Indeed', 'LinkedIn', 'LinkedIn',
'Google Search', 'Website', 'Indeed', 'LinkedIn', 'Indeed',
'Indeed', 'Google Search', 'Indeed', 'LinkedIn', 'Indeed',
'Diversity Job Fair', 'Google Search', 'Indeed', 'Indeed', 'Other',
'Indeed', 'Indeed', 'LinkedIn', 'CareerBuilder',
'Diversity Job Fair', 'LinkedIn', 'Employee Referral', 'Indeed',
'LinkedIn', 'Employee Referral', 'Website', 'Employee Referral',
'Google Search', 'LinkedIn', 'Employee Referral',
'Diversity Job Fair', 'CareerBuilder', 'Indeed',
'Employee Referral', 'LinkedIn', 'Website', 'Google Search',
'Diversity Job Fair', 'LinkedIn', 'LinkedIn', 'LinkedIn',
'Google Search', 'Employee Referral', 'Employee Referral',
'LinkedIn'], dtype=object)
```

# Построение графика "Ящик с усами (boxplot)":

Отображает одномерное распределение вероятности. Построение графика для колонки данных "EngagementSurvey".

In [16]:
```
#по горизонтали
sns.boxplot(x=data['EngagementSurvey'])
```

Out[16]: <AxesSubplot:xlabel='EngagementSurvey'>




In [17]:
```
# по вертикали
sns.boxplot(y=data['EngagementSurvey'])
```

Out[17]: <AxesSubplot:ylabel='EngagementSurvey'>