Министерство науки и высшего образования 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)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления

Рубежный контроль №1
Вариант 22

По дисциплине:
«Технологии машинного обучения»

Выполнил:

Студент группы ИУ5-62 _____ **Чепкин Д.А.**

(Подпись, дата) (Фамилия И.О.)

Проверил:

_____ **Гапанюк Ю. Е.**

(Подпись, дата) (Фамилия И.О.)

Москва, 2021

1. Задание

Номер варианта	Номер задачи	Номер набора данных, указанного в задаче
22	3	6

Дополнительные требования по группам:

Для студентов групп ИУ5-62Б, ИУ5Ц-82Б - для произвольной колонки данных построить гистограмму.

Задача №3.

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?

Набор данных:

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

Выполнение задания

Импорт датасета и масштабирование данных

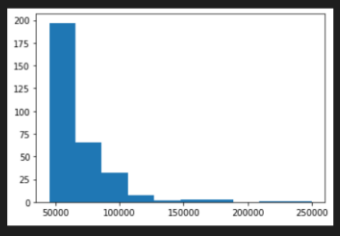
```
[1]
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

[34]
data = pd.read_csv('HRDataset_v14.csv',sep=",")

[35]
data.head()
```

	Employee_Name	EmpID	MarriedID	MaritalStatusID	GenderID	EmpStatusID	DeptID	PerfScoreID	FromDiversityJobFairID	Salary	...	ManagerName	ManagerID	RecruitmentSource	PerformanceScore	EngagementSurvey
0	Adinolfi, Wilson K	10026	0	0	1	1	5	4	0	62506	...	Michael Albert	22.0	LinkedIn	Exceeds	4.60
1	Ait Sidi, Karthikeyan	10084	1	1	1	5	3	3	0	104437	...	Simon Roup	4.0	Indeed	Fully Meets	4.96
2	Akinkuolie, Sarah	10196	1	1	0	5	5	3	0	64955	...	Kissy Sullivan	20.0	LinkedIn	Fully Meets	3.02
3	Alagbe,Trina	10088	1	1	0	1	5	3	0	64991	...	Elijiah Gray	16.0	Indeed	Fully Meets	4.84
4	Anderson, Carol	10069	0	2	0	5	5	3	0	50825	...	Webster Butler	39.0	Google Search	Fully Meets	5.00

5 rows × 36 columns

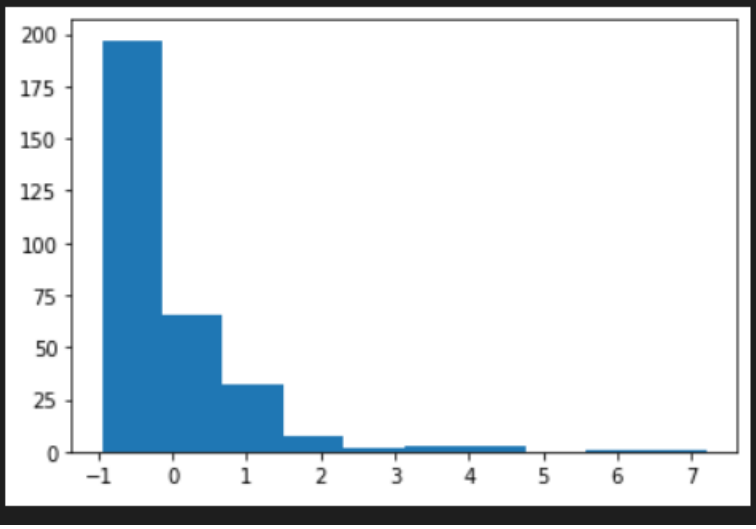
Для масштабирования данных выберем Salary

```
5 rows × 36 columns

[19]
data.shape

(311, 36)

[20]
plt.hist(data.Salary)
plt.show()
```



[21]
```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
scaled_total_salary = scaler.fit_transform(data[['Salary']])
```

[22]
```
plt.hist(scaled_total_salary)
plt.show()
```



Преобразование категориальных признаков

Произведем категориальное преобразование для столбцов RaceDesc и Position методами one hot encoding и label encoding соответственно.

[27]
```
data.RaceDesc.unique()
```

```
array(['White', 'Black or African American', 'Two or more races', 'Asian',
       'American Indian or Alaska Native', 'Hispanic'], dtype=object)
```

[23]
```
data.Position.unique()
```

```
array(['Production Technician I', 'Sr. DBA', 'Production Technician II',
       'Software Engineer', 'IT Support', 'Data Analyst',
       'Database Administrator', 'Enterprise Architect', 'Sr. Accountant',
       'Production Manager', 'Accountant I', 'Area Sales Manager',
       'Software Engineering Manager', 'BI Director',
       'Director of Operations', 'Sr. Network Engineer', 'Sales Manager',
       'BI Developer', 'IT Manager - Support', 'Network Engineer',
       'IT Director', 'Director of Sales', 'Administrative Assistant',
       'President & CEO', 'Senior BI Developer',
       'Shared Services Manager', 'IT Manager - Infra',
       'Principal Data Architect', 'Data Architect', 'IT Manager - DB',
       'Data Analyst ', 'CIO'], dtype=object)
```

Для метода one hote encoding можно использовать методы библиотеки scklearn или pandas.

[28]
```
data_race = pd.get_dummies(data=data.RaceDesc)
data_race.head()
```

	American Indian or Alaska Native	Asian	Black or African American	Hispanic	Two or more races	White
0	0	0	0	0	0	1
1	0	0	0	0	0	1
2	0	0	0	0	0	1
3	0	0	0	0	0	1
4	0	0	0	0	0	1

LabelEncoder

[40]
```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
position_le = le.fit_transform(data.Position)
data.Position.unique()
```

```
array(['Production Technician I', 'Sr. DBA', 'Production Technician II',
       'Software Engineer', 'IT Support', 'Data Analyst',
       'Database Administrator', 'Enterprise Architect', 'Sr. Accountant',
       'Production Manager', 'Accountant I', 'Area Sales Manager',
       'Software Engineering Manager', 'BI Director',
       'Director of Operations', 'Sr. Network Engineer', 'Sales Manager',
       'BI Developer', 'IT Manager - Support', 'Network Engineer',
       'IT Director', 'Director of Sales', 'Administrative Assistant',
       'President & CEO', 'Senior BI Developer',
       'Shared Services Manager', 'IT Manager - Infra',
       'Principal Data Architect', 'Data Architect', 'IT Manager - DB',
       'Data Analyst ', 'CIO'], dtype=object)
```

[41]
```
np.unique(position_le)
```

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31])
```

```
[42]
le.inverse_transform([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31])

array(['Accountant I', 'Administrative Assistant', 'Area Sales Manager',
       'BI Developer', 'BI Director', 'CIO', 'Data Analyst',
       'Data Analyst ', 'Data Architect', 'Database Administrator',
       'Director of Operations', 'Director of Sales',
       'Enterprise Architect', 'IT Director', 'IT Manager - DB',
       'IT Manager - Infra', 'IT Manager - Support', 'IT Support',
       'Network Engineer', 'President & CEO', 'Principal Data Architect',
       'Production Manager', 'Production Technician I',
       'Production Technician II', 'Sales Manager', 'Senior BI Developer',
       'Shared Services Manager', 'Software Engineer',
       'Software Engineering Manager', 'Sr. Accountant', 'Sr. DBA',
       'Sr. Network Engineer'], dtype=object)
```

Дополнительное требование:

Гистограмма для столбца Absence

```
[43]
plt.hist(data.Absences)
plt.show()
```