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 (ИУ5)_____

ОТЧЕТ
по лабораторной работе

«Обработка пропусков в данных, кодирование категориальных признаков, масштабирование данных.»

ДИСЦИПЛИНА: «Технологии машинного обучения»

Выполнил: студент гр. _ИУ5-62Б_ _____ (_____Кудрявцев С.Д.___)
(Подпись) (Ф.И.О.)

Проверил: _____ (_____Гапанюк Ю.Е.___)
(Подпись) (Ф.И.О.)

2020 г.

Лабораторная работа №3

Обработка пропусков в данных, кодирование категориал масштабирование данных

Цель лабораторной работы

Задание

1. Выбрать набор данных (датасет), содержащий категориальные признаки и пропуски в пунктов можно использовать несколько различных наборов данных (один для обрабо категориальных признаков и т.д.)
2. Для выбранного датасета (датасетов) на основе материалов лекции решить следующ
   - обработку пропусков в данных;
   - кодирование категориальных признаков;
   - масштабирование данных.

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

Загрузка данных

Ссылка на датасет: https://www.kaggle.com/fivethirtyeight/fivethirtyeight-comic-characters-data

```
data = pd.read_csv('data/marvel-wikia-data.csv', sep=",")
data.head()
```

	page_id	name	urlslug
0	1678	Spider-Man (Peter Parker)	\/Spider-Man_(Peter_Parker)
1	7139	Captain America (Steven Rogers)	\/Captain_America_(Steven_Rogers)
2	64786	Wolverine (James \"Logan\" Howlett)	\/Wolverine_(James_%22Logan%22_Howlett)

3	1868	Iron Man (Anthony \"Tony\" Stark)	\/Iron_Man_(Anthony_%22Tony%22_Stark)
4	2460	Thor (Thor Odinson)	\/Thor_(Thor_Odinson) N

```
# размер набора данных
data.shape
```

```
(16376, 13)
```

```
# типы колонок
data.dtypes
```

```
page_id              int64
name                object
urlslug             object
ID                  object
ALIGN               object
EYE                 object
HAIR                object
SEX                 object
GSM                 object
ALIVE               object
APPEARANCES        float64
FIRST APPEARANCE    object
Year               float64
dtype: object
```

```
# проверим есть ли пропущенные значения
data.isnull().sum()
```

```
page_id                0
name                   0
urlslug                0
ID                  3770
ALIGN               2812
EYE                 9767
HAIR                4264
SEX                  854
GSM                16286
ALIVE                  3
APPEARANCES         1096
FIRST APPEARANCE     815
Year                 815
dtype: int64
```

```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 16376
```

1. Обработка пропусков в данных

Простые стратегии: удаление

```
data = data.dropna(axis=1, thresh=int(data.shape[0] * 0.49))
```

```
data.isnull().sum()
```

```
page_id                0
name                   0
urlslug                0
ID                  3770
ALIGN               2812
HAIR                4264
SEX                  854
ALIVE                  3
APPEARANCES         1096
FIRST APPEARANCE     815
Year                 815
dtype: int64
```

```
data[data['ALIVE'].isnull()]
```

	page_id	name	urlslug	ID	ALIGN	HAIR	SEX	ALIVE	APPEAR
16293	541449	Mj7711	\/User:Mj7711	NaN	NaN	NaN	NaN	NaN	
16329	714409	Sharjeel786	\/User:Sharjeel786	NaN	NaN	NaN	NaN	NaN	
16347	462671	TOR\/test	\/User:TOR\/test	NaN	NaN	NaN	NaN	NaN	

```
# Удаление 3 строк
data = data.drop(data.index[[16293,16329,16347]])
```

```
data.isnull().sum()
```

```
page_id                0
name                   0
urlslug                0
ID                  3767
ALIGN               2809
HAIR                4261
SEX                  851
ALIVE                  0
APPEARANCES         1093
FIRST APPEARANCE     812
Year                 812
dtype: int64
```

"Внедрение значений" - импьютация (imputation)

Обработка пропусков в числовых данных

↳ 20 cells hidden

Обработка пропусков в категориальных данных

4 6 cells hidden

2. Преобразование категориальных признаков в числов

```
cat_enc = pd.DataFrame({'c1':data_imp2.T[0]})
cat_enc
```

	c1
0	Male Characters
1	Male Characters
2	Male Characters
3	Male Characters
4	Male Characters
...	...
16368	Male Characters
16369	Male Characters
16370	Male Characters
16371	Male Characters
16372	Male Characters

16373 rows × 1 columns

Кодирование категорий целочисленными значениями - label en

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder

le = LabelEncoder()
cat_enc_le = le.fit_transform(cat_enc['c1'])
```

```
cat_enc['c1'].unique()
```

```
array(['Male Characters', 'Female Characters', 'Genderfluid Characters',
       'Agender Characters'], dtype=object)
```

```
np.unique(cat_enc_le)
```

```
array([0, 1, 2, 3])
```

array([0, 1, 2, 3])

```
le.inverse_transform([0, 1, 2, 3])
```

```
array(['Agender Characters', 'Female Characters',
       'Genderfluid Characters', 'Male Characters'], dtype=object)
```

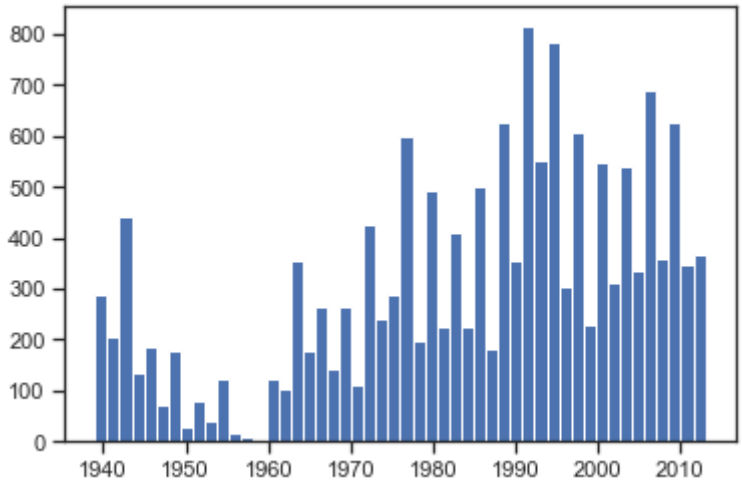
3. Масштабирование данных

```
from sklearn.preprocessing import MinMaxScaler, StandardScaler, Normalizer
```

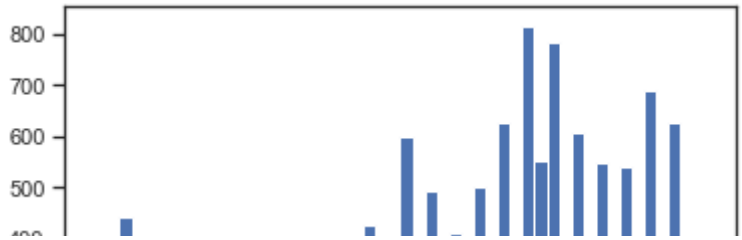
```
sc1 = MinMaxScaler()
sc1_data = sc1.fit_transform(data[['Year']])
```

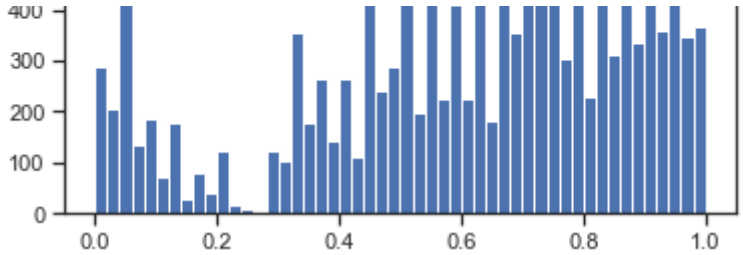
```
plt.hist(data['Year'], 50)
plt.show()
```

```
c:\users\user\appdata\local\programs\python\python37-32\lib\site-packages\nump
  keep = (tmp_a >= first_edge)
c:\users\user\appdata\local\programs\python\python37-32\lib\site-packages\nump
  keep &= (tmp_a <= last_edge)
```




```
plt.hist(sc1_data, 50)
plt.show()
```




400
300
200
100
0
0.0
0.2
0.4
0.6
0.8
1.0