Московский государственный технический

университет им. Н.Э. Баумана.

Факультет «Информатика и управление»

Кафедра ИУ5. Курс «Технологии машинного обучения»

Отчет по лабораторной работе № 2

Выполнил:
студент группы ИУ5-61Б
Егор Утробин
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.
Подпись и дата:

Москва, 2021 г.

lab2_1

March 22, 2021

1 №2

1.0.1

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from pandas.plotting import scatter_matrix
import warnings
warnings.filterwarnings('ignore')
sns.set(style="ticks")
%matplotlib inline
```

[2]:
```
data = pd.read_csv('country_vaccinations.csv')
```

[3]:
```
data.head()
```

[3]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred
0	Albania	ALB	2021-01-10	0.0	0.0	NaN	NaN	NaN	0.00	0.00
1	Albania	ALB	2021-01-11	NaN	NaN	NaN	NaN	64.0	NaN	NaN
2	Albania	ALB	2021-01-12	128.0	128.0	NaN	NaN	64.0	0.00	0.00
3	Albania	ALB	2021-01-13	188.0	188.0	NaN	60.0	63.0	0.01	0.01
4	Albania	ALB	2021-01-14	266.0	266.0	NaN	78.0	66.0	0.01	0.01

```
   people_fully_vaccinated_per_hundred  daily_vaccinations_per_million  \
0                                  NaN                             NaN
1                                  NaN                            22.0
2                                  NaN                            22.0
3                                  NaN                            22.0
4                                  NaN                            23.0

          vaccines         source_name  \
0  Pfizer/BioNTech  Ministry of Health
1  Pfizer/BioNTech  Ministry of Health
2  Pfizer/BioNTech  Ministry of Health
3  Pfizer/BioNTech  Ministry of Health
4  Pfizer/BioNTech  Ministry of Health

                                      source_website
0  https://shendetesia.gov.al/covid19-ministria-e…
1  https://shendetesia.gov.al/covid19-ministria-e…
2  https://shendetesia.gov.al/covid19-ministria-e…
3  https://shendetesia.gov.al/covid19-ministria-e…
4  https://shendetesia.gov.al/covid19-ministria-e…
```

[4]:
```
data.dtypes
```

[4]:
```
country                                object
iso_code                               object
date                                   object
total_vaccinations                    float64
people_vaccinated                     float64
people_fully_vaccinated               float64
daily_vaccinations_raw                float64
daily_vaccinations                    float64
total_vaccinations_per_hundred        float64
people_vaccinated_per_hundred         float64
people_fully_vaccinated_per_hundred   float64
daily_vaccinations_per_million        float64
vaccines                               object
source_name                            object
source_website                         object
dtype: object
```

[5]:
```
data.isnull().sum()
#
```

[5]:
```
country                                0
iso_code                             272
date                                   0
```

```
total_vaccinations                    1214
people_vaccinated                     1615
people_fully_vaccinated               2277
daily_vaccinations_raw                1583
daily_vaccinations                     135
total_vaccinations_per_hundred        1214
people_vaccinated_per_hundred         1615
people_fully_vaccinated_per_hundred   2277
daily_vaccinations_per_million         135
vaccines                                 0
source_name                              0
source_website                           0
dtype: int64
```

[6]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3555 entries, 0 to 3554
Data columns (total 15 columns):
 #   Column                               Non-Null Count  Dtype
---  ------                               --------------  -----
 0   country                              3555 non-null   object
 1   iso_code                             3283 non-null   object
 2   date                                 3555 non-null   object
 3   total_vaccinations                   2341 non-null   float64
 4   people_vaccinated                    1940 non-null   float64
 5   people_fully_vaccinated              1278 non-null   float64
 6   daily_vaccinations_raw               1972 non-null   float64
 7   daily_vaccinations                   3420 non-null   float64
 8   total_vaccinations_per_hundred       2341 non-null   float64
 9   people_vaccinated_per_hundred        1940 non-null   float64
 10  people_fully_vaccinated_per_hundred  1278 non-null   float64
 11  daily_vaccinations_per_million       3420 non-null   float64
 12  vaccines                             3555 non-null   object
 13  source_name                          3555 non-null   object
 14  source_website                       3555 non-null   object
dtypes: float64(9), object(6)
memory usage: 416.7+ KB
```

1.0.2

[7]:
```
#          ,
data.drop(['source_name','source_website'], axis = 1, inplace = True)
```

[8]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3555 entries, 0 to 3554
```

```
Data columns (total 13 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   country                               3555 non-null   object
 1   iso_code                              3283 non-null   object
 2   date                                  3555 non-null   object
 3   total_vaccinations                    2341 non-null   float64
 4   people_vaccinated                     1940 non-null   float64
 5   people_fully_vaccinated               1278 non-null   float64
 6   daily_vaccinations_raw                1972 non-null   float64
 7   daily_vaccinations                    3420 non-null   float64
 8   total_vaccinations_per_hundred        2341 non-null   float64
 9   people_vaccinated_per_hundred         1940 non-null   float64
 10  people_fully_vaccinated_per_hundred   1278 non-null   float64
 11  daily_vaccinations_per_million        3420 non-null   float64
 12  vaccines                              3555 non-null   object
dtypes: float64(9), object(4)
memory usage: 361.2+ KB
```

[9]:
```
#
data['total_vaccinations'] = data['total_vaccinations'].replace(0,np.nan)
data['total_vaccinations'] = data['total_vaccinations'].
 ↪fillna(data['total_vaccinations'].mean())
```

[10]:
```
data.head()
```

[10]:

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	daily_vaccinations	total_vaccinations_per_hundred	people_vaccinated_per_hundred
0	Albania	ALB	2021-01-10	1.508878e+06	0.0	NaN	NaN	NaN	0.00	0.00
1	Albania	ALB	2021-01-11	1.508878e+06	NaN	NaN	NaN	64.0	NaN	NaN
2	Albania	ALB	2021-01-12	1.280000e+02	128.0	NaN	NaN	64.0	0.00	0.00
3	Albania	ALB	2021-01-13	1.880000e+02	188.0	NaN	60.0	63.0	0.01	0.01
4	Albania	ALB	2021-01-14	2.660000e+02	266.0	NaN	78.0	66.0	0.01	0.01

```
   people_fully_vaccinated_per_hundred  daily_vaccinations_per_million  \
0                                  NaN                             NaN
1                                  NaN                            22.0
2                                  NaN                            22.0
3                                  NaN                            22.0
4                                  NaN                            23.0

          vaccines
0  Pfizer/BioNTech
1  Pfizer/BioNTech
2  Pfizer/BioNTech
3  Pfizer/BioNTech
4  Pfizer/BioNTech
```

[11]:
```
data.isnull().sum()
#
```

[11]:
```
country                                0
iso_code                             272
date                                   0
total_vaccinations                     0
people_vaccinated                   1615
people_fully_vaccinated             2277
daily_vaccinations_raw              1583
daily_vaccinations                   135
total_vaccinations_per_hundred      1214
people_vaccinated_per_hundred       1615
people_fully_vaccinated_per_hundred 2277
daily_vaccinations_per_million       135
vaccines                               0
dtype: int64
```

lab2_2

March 22, 2021

1 №2

1.0.1

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.impute import SimpleImputer
```

[2]:
```
data = pd.read_csv('country_wise_latest.csv')
```

[3]:
```
data.head()
```

[3]:

	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	1 week change	1 week % increase	WHO Region
0	Afghanistan	36263	1269	25198	9796	106	10	18	3.50	69.49	5.04	35526	737	2.07	Eastern Mediterranean
1	Albania	4880	144	2745	1991	117	6	63	2.95	56.25	5.25	4171	709		
2	Algeria	27973	1163	18837	7973	616	8	749	4.16	67.34	6.17	23691	4282		
3	Andorra	907	52	803	52	10	0	0	5.73	88.53	6.48	884	23		
4	Angola	950	41	242	667	18	1	0	4.32	25.47	16.94	749	201		

```
1              17.00                Europe
2              18.07                Africa
3               2.60                Europe
4              26.84                Africa
```

[4]:
```
data['WHO Region'].value_counts()
```

[4]:
```
Europe                  56
Africa                  48
Americas                35
Eastern Mediterranean   22
Western Pacific         16
South-East Asia         10
Name: WHO Region, dtype: int64
```

[5]:
```
#            Recovered   WHO Region
data = pd.get_dummies(data, columns=['Recovered','WHO Region'])
```

[6]:
```
data.head()
```

[6]:
```
  Country/Region  Confirmed  Deaths  Active  New cases  New deaths  \
0    Afghanistan      36263    1269    9796        106          10
1        Albania       4880     144    1991        117           6
2        Algeria      27973    1163    7973        616           8
3        Andorra        907      52      52         10           0
4         Angola        950      41     667         18           1

   New recovered  Deaths / 100 Cases  Recovered / 100 Cases  \
0             18                3.50                  69.49
1             63                2.95                  56.25
2            749                4.16                  67.34
3              0                5.73                  88.53
4              0                4.32                  25.47

   Deaths / 100 Recovered  …  Recovered_602249  Recovered_951166  \
0                    5.04  …                 0                 0
1                    5.25  …                 0                 0
2                    6.17  …                 0                 0
3                    6.48  …                 0                 0
4                   16.94  …                 0                 0

   Recovered_1325804  Recovered_1846641  WHO Region_Africa  \
0                  0                  0                  0
1                  0                  0                  0
2                  0                  0                  1
3                  0                  0                  0
4                  0                  0                  1
```

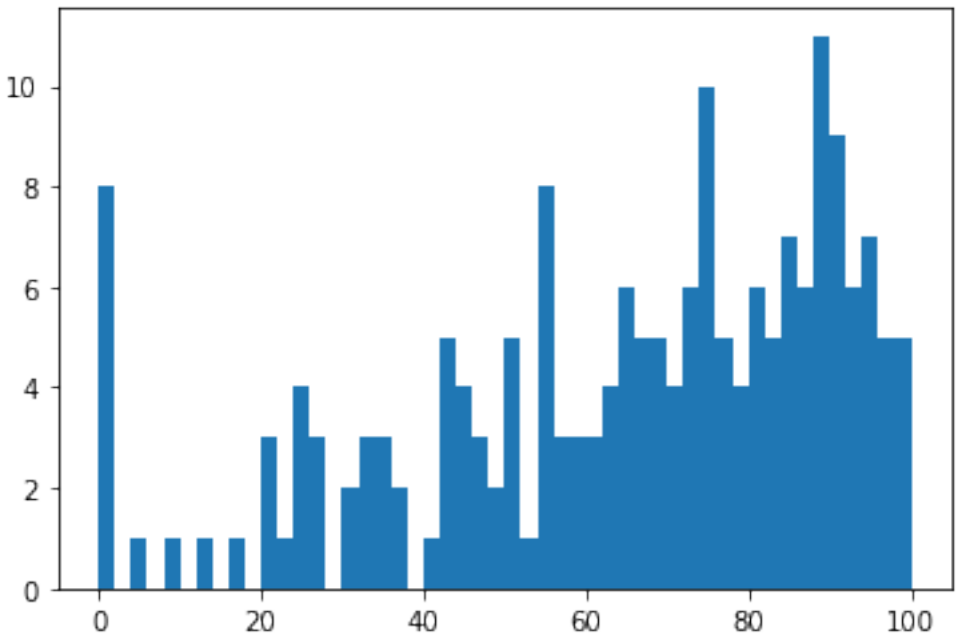
```
   WHO Region_Americas  WHO Region_Eastern Mediterranean  WHO Region_Europe  \
0                    0                                 1                  0
1                    0                                 0                  1
2                    0                                 0                  0
3                    0                                 0                  1
4                    0                                 0                  0

   WHO Region_South-East Asia  WHO Region_Western Pacific
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

[5 rows x 197 columns]
```

1.0.2

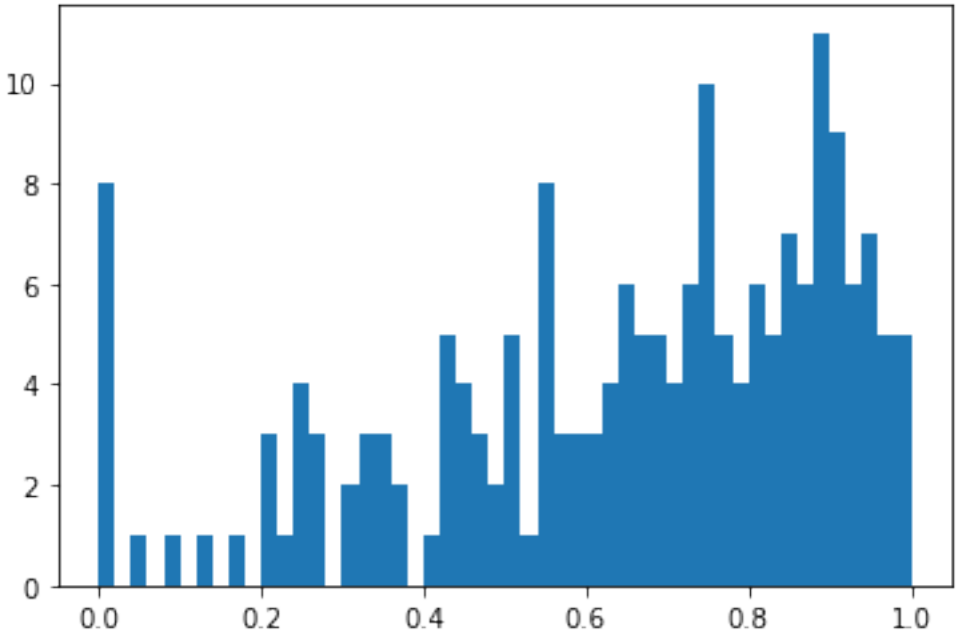
[7]:
```
from sklearn.preprocessing import StandardScaler, MinMaxScaler, StandardScaler,␣
↪Normalizer
```

[8]:
```
sc1 = MinMaxScaler()
sc1_data = sc1.fit_transform(data[['Recovered / 100 Cases']])
```

[9]:
```
plt.hist(data['Recovered / 100 Cases'], 50)
plt.show()
```



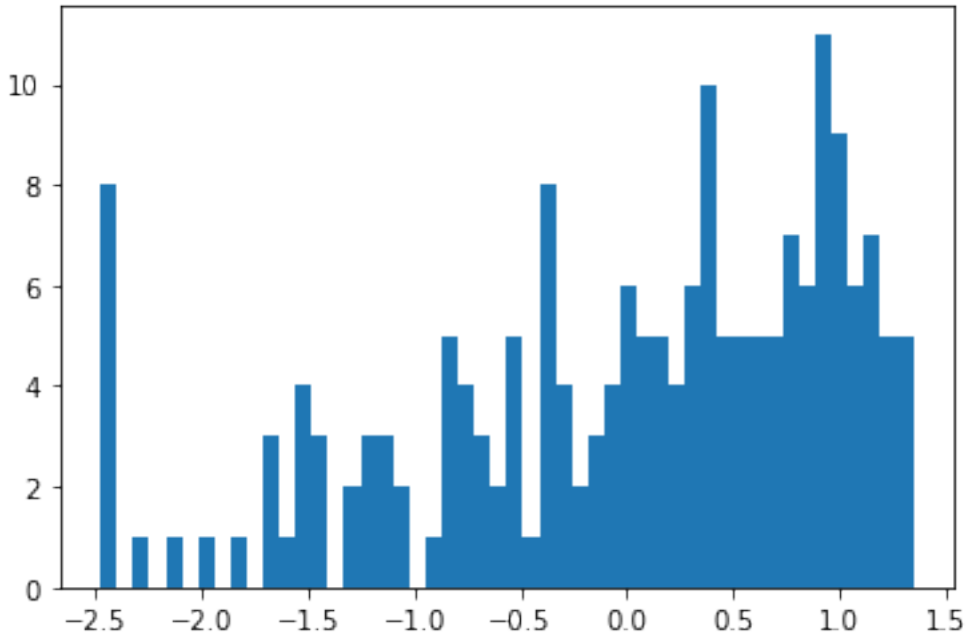
[10]:
```
plt.hist(sc1_data, 50)
plt.show()
```



[11]:
```
#          ,
data.drop(['Country/Region','New deaths','New recovered'], axis = 1, inplace =␣
 ↪True)
```

[12]:
```
sc2 = StandardScaler()
sc2_data = sc2.fit_transform(data[['Recovered / 100 Cases']])
```

[13]:
```
plt.hist(sc2_data, 50)
plt.show()
```



[14]:
```
data.head()
```

[14]:

	Confirmed	Deaths	Active	New cases	Deaths / 100 Cases	Recovered / 100 Cases	Deaths / 100 Recovered	Confirmed last week	1 week change	1 week % increase	…	Recovered_602249	Recovered_951166	Recovered_1325804	Recovered_1846641	WHO Region_Africa
0	36263	1269	9796	106	3.50	69.49	5.04	35526	737	2.07	…	0	0	0	0	0
1	4880	144	1991	117	2.95	56.25	5.25	4171	709	17.00	…	0	0	0	0	0
2	27973	1163	7973	616	4.16	67.34	6.17	23691	4282	18.07	…	0	0	0	0	0
3	907	52	52	10	5.73	88.53	6.48	884	23	2.60	…	0	0	0	0	0
4	950	41	667	18	4.32	25.47	16.94	749	201	26.84	…	0	0	0	0	0

```
2                   0                   0                   1
3                   0                   0                   0
4                   0                   0                   1

   WHO Region_Americas  WHO Region_Eastern Mediterranean  WHO Region_Europe  \
0                    0                                 1                  0
1                    0                                 0                  1
2                    0                                 0                  0
3                    0                                 0                  1
4                    0                                 0                  0

   WHO Region_South-East Asia  WHO Region_Western Pacific
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

[5 rows x 194 columns]
```