Буклин С. В. ИУ5-21М

РК №1 по курсу "Методы иашинного обучения"

Вариант №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/karangadiya/fifa19 (https://www.kaggle.com/karangadiya/fifa19)

In [0]:

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

In [0]:

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6
bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.googleusercontent.com&redirect_uri=urn%3Aietf%3Awg%3Aoauth
%3A2.0%3Aoob&scope=email%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fdocs.test%20https%3A%2F%2F
www.googleapis.com%2Fauth%2Fdrive%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fdrive.photos.read
only%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fpeopleapi.readonly&response_type=code

Enter your authorization code:
··········
Mounted at /content/drive
```

In [0]:

```
data = pd.read_csv('/content/drive/My Drive/Colab Notebooks/data.csv', sep=',')
```

In [0]:

```
data.head()
```

Out[0]:

	Unnamed: 0	ID	Name	Age	Photo	Nationality	
0	0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://cdn.sofifa.org/flag
1	1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://cdn.sofifa.org/flag
2	2	190871	Neymar Jr	26	https://cdn.sofifa.org/players/4/19/190871.png	Brazil	https://cdn.sofifa.org/flag
3	3	193080	De Gea	27	https://cdn.sofifa.org/players/4/19/193080.png	Spain	https://cdn.sofifa.org/flag
4	4	192985	K. De Bruyne	27	https://cdn.sofifa.org/players/4/19/192985.png	Belgium	https://cdn.sofifa.org/fla

5 rows × 89 columns

In [0]:

```
data.shape
```

Out[0]:

```
(18207, 89)
```

In [0]:

```
data.dtypes
```

Out[0]:

```
Unnamed: 0                    int64
ID                            int64
Name                         object
Age                           int64
Photo                        object
Nationality                  object
Flag                         object
Overall                       int64
Potential                     int64
Club                         object
Club Logo                    object
Value                        object
Wage                         object
Special                       int64
Preferred Foot               object
International Reputation    float64
Weak Foot                   float64
Skill Moves                 float64
Work Rate                    object
Body Type                    object
Real Face                    object
Position                     object
Jersey Number               float64
Joined                       object
Loaned From                  object
Contract Valid Until         object
Height                       object
Weight                       object
LS                           object
ST                           object
                             ...
Dribbling                   float64
Curve                       float64
FKAccuracy                  float64
LongPassing                 float64
BallControl                 float64
Acceleration                float64
SprintSpeed                 float64
Agility                     float64
Reactions                   float64
Balance                     float64
ShotPower                   float64
Jumping                     float64
Stamina                     float64
Strength                    float64
LongShots                   float64
Aggression                  float64
Interceptions               float64
Positioning                 float64
Vision                      float64
Penalties                   float64
Composure                   float64
Marking                     float64
StandingTackle              float64
SlidingTackle               float64
GKDiving                    float64
GKHandling                  float64
GKKicking                   float64
GKPositioning               float64
GKReflexes                  float64
Release Clause               object
Length: 89, dtype: object
```

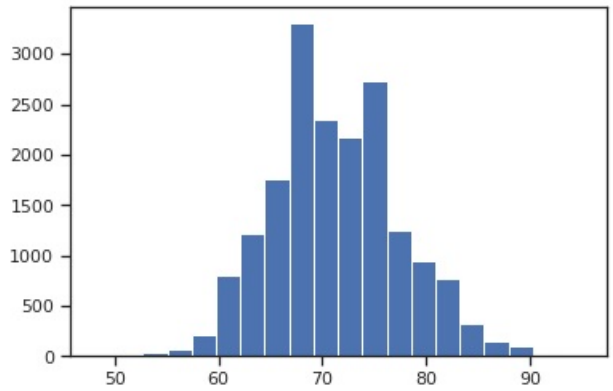
Масштабирование данных

In [0]:

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

In [0]:

```
plt.hist(data['Potential'], 20)
plt.show()
```



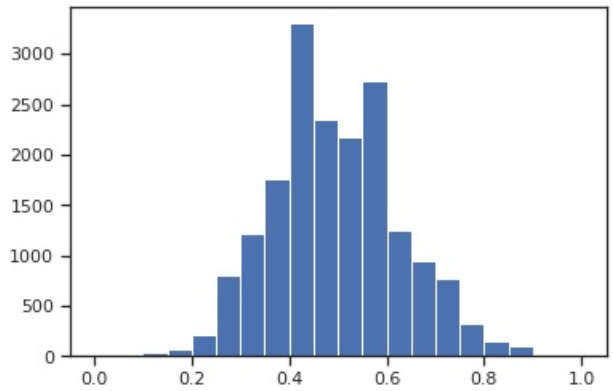
In [0]:

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

```
/usr/local/lib/python3.6/dist-packages/sklearn/preprocessing/data.py:334: DataConversionWarning
: Data with input dtype int64 were all converted to float64 by MinMaxScaler.
  return self.partial_fit(X, y)
```

In [0]:

```
plt.hist(sc1_data, 20)
plt.show()
```



Преобразование категориальных признаков в количественные

In [0]:

```
cat_temp_data = data[['Nationality']]
cat_temp_data.tail(10)
```

Out[0]:

	Nationality
18197	Republic of Ireland
18198	England
18199	Canada
18200	Scotland
18201	Republic of Ireland
18202	England
18203	Sweden
18204	England
18205	England
18206	England

In [0]:

```
cat_temp_data['Nationality'].unique()
```

Out[0]:

```
array(['Argentina', 'Portugal', 'Brazil', 'Spain', 'Belgium', 'Croatia',
       'Uruguay', 'Slovenia', 'Poland', 'Germany', 'France', 'England',
       'Italy', 'Egypt', 'Colombia', 'Denmark', 'Gabon', 'Wales',
       'Senegal', 'Costa Rica', 'Slovakia', 'Netherlands',
       'Bosnia Herzegovina', 'Morocco', 'Serbia', 'Algeria', 'Austria',
       'Greece', 'Chile', 'Sweden', 'Korea Republic', 'Finland', 'Guinea',
       'Montenegro', 'Armenia', 'Switzerland', 'Norway', 'Czech Republic',
       'Scotland', 'Ghana', 'Central African Rep.', 'DR Congo',
       'Ivory Coast', 'Russia', 'Ukraine', 'Iceland', 'Mexico', 'Jamaica',
       'Albania', 'Venezuela', 'Japan', 'Turkey', 'Ecuador', 'Paraguay',
       'Mali', 'Nigeria', 'Cameroon', 'Dominican Republic', 'Israel',
       'Kenya', 'Hungary', 'Republic of Ireland', 'Romania',
       'United States', 'Cape Verde', 'Australia', 'Peru', 'Togo',
       'Syria', 'Zimbabwe', 'Angola', 'Burkina Faso', 'Iran', 'Estonia',
       'Tunisia', 'Equatorial Guinea', 'New Zealand', 'FYR Macedonia',
       'United Arab Emirates', 'China PR', 'Guinea Bissau', 'Bulgaria',
       'Kosovo', 'South Africa', 'Madagascar', 'Georgia', 'Tanzania',
       'Gambia', 'Cuba', 'Belarus', 'Uzbekistan', 'Benin', 'Congo',
       'Mozambique', 'Honduras', 'Canada', 'Northern Ireland', 'Cyprus',
       'Saudi Arabia', 'Curacao', 'Moldova', 'Bolivia',
       'Trinidad & Tobago', 'Sierra Leone', 'Zambia', 'Chad',
       'Philippines', 'Haiti', 'Comoros', 'Libya', 'Panama',
       'São Tomé & Príncipe', 'Eritrea', 'Oman', 'Iraq', 'Burundi',
       'Fiji', 'New Caledonia', 'Lithuania', 'Luxembourg', 'Korea DPR',
       'Liechtenstein', 'St Kitts Nevis', 'Latvia', 'Suriname', 'Uganda',
       'El Salvador', 'Bermuda', 'Kuwait', 'Antigua & Barbuda',
       'Thailand', 'Mauritius', 'Guatemala', 'Liberia', 'Kazakhstan',
       'Niger', 'Mauritania', 'Montserrat', 'Namibia', 'Azerbaijan',
       'Guam', 'Faroe Islands', 'India', 'Nicaragua', 'Barbados',
       'Lebanon', 'Palestine', 'Guyana', 'Sudan', 'St Lucia', 'Ethiopia',
       'Puerto Rico', 'Grenada', 'Jordan', 'Rwanda', 'Qatar',
       'Afghanistan', 'Hong Kong', 'Andorra', 'Malta', 'Belize',
       'South Sudan', 'Indonesia', 'Botswana'], dtype=object)
```

In [0]:

```
cat_temp_data[cat_temp_data['Nationality'].isnull()].shape
```

Out[0]:

```
(0, 1)
```

In [0]:

```
from sklearn.impute import SimpleImputer
```

In [0]:

```
imp2 = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
data_imp2 = imp2.fit_transform(cat_temp_data)
data_imp2
```

Out[0]:

```
array([['Argentina'],
       ['Portugal'],
       ['Brazil'],
       ...,
       ['England'],
       ['England'],
       ['England']], dtype=object)
```

In [0]:

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

Out[0]:

	c1
0	Argentina
1	Portugal
2	Brazil
3	Spain
4	Belgium
5	Belgium
6	Croatia
7	Uruguay
8	Spain
9	Slovenia
10	Poland
11	Germany
12	Uruguay
13	Spain
14	France
15	Argentina
16	England
17	France
18	Germany
19	Belgium
20	Spain
21	Uruguay
22	Germany
23	Argentina
24	Italy
25	France
26	Egypt
27	Brazil
28	Colombia
29	Italy
...	...
18177	Republic of Ireland
18178	Sweden
18179	England
18180	Scotland

18181	Republic of Ireland
18182	Colombia
18183	England
18184	England
18185	Republic of Ireland
18186	China PR
18187	Germany
18188	Wales
18189	Germany
18190	England
18191	England
18192	England
18193	Chile
18194	Italy
18195	Republic of Ireland
18196	Japan
18197	Republic of Ireland
18198	England
18199	Canada
18200	Scotland
18201	Republic of Ireland
18202	England
18203	Sweden
18204	England
18205	England
18206	England

18207 rows × 1 columns

Label encoding

In [0]:

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

In [0]:

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

In [0]:

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

Out[0]:

```
array(['Argentina', 'Portugal', 'Brazil', 'Spain', 'Belgium', 'Croatia',
       'Uruguay', 'Slovenia', 'Poland', 'Germany', 'France', 'England',
       'Italy', 'Egypt', 'Colombia', 'Denmark', 'Gabon', 'Wales',
       'Senegal', 'Costa Rica', 'Slovakia', 'Netherlands',
       'Bosnia Herzegovina', 'Morocco', 'Serbia', 'Algeria', 'Austria',
       'Greece', 'Chile', 'Sweden', 'Korea Republic', 'Finland', 'Guinea',
       'Montenegro', 'Armenia', 'Switzerland', 'Norway', 'Czech Republic',
       'Scotland', 'Ghana', 'Central African Rep.', 'DR Congo',
       'Ivory Coast', 'Russia', 'Ukraine', 'Iceland', 'Mexico', 'Jamaica',
       'Albania', 'Venezuela', 'Japan', 'Turkey', 'Ecuador', 'Paraguay',
       'Mali', 'Nigeria', 'Cameroon', 'Dominican Republic', 'Israel',
       'Kenya', 'Hungary', 'Republic of Ireland', 'Romania',
       'United States', 'Cape Verde', 'Australia', 'Peru', 'Togo',
       'Syria', 'Zimbabwe', 'Angola', 'Burkina Faso', 'Iran', 'Estonia',
       'Tunisia', 'Equatorial Guinea', 'New Zealand', 'FYR Macedonia',
       'United Arab Emirates', 'China PR', 'Guinea Bissau', 'Bulgaria',
       'Kosovo', 'South Africa', 'Madagascar', 'Georgia', 'Tanzania',
       'Gambia', 'Cuba', 'Belarus', 'Uzbekistan', 'Benin', 'Congo',
       'Mozambique', 'Honduras', 'Canada', 'Northern Ireland', 'Cyprus',
       'Saudi Arabia', 'Curacao', 'Moldova', 'Bolivia',
       'Trinidad & Tobago', 'Sierra Leone', 'Zambia', 'Chad',
       'Philippines', 'Haiti', 'Comoros', 'Libya', 'Panama',
       'São Tomé & Príncipe', 'Eritrea', 'Oman', 'Iraq', 'Burundi',
       'Fiji', 'New Caledonia', 'Lithuania', 'Luxembourg', 'Korea DPR',
       'Liechtenstein', 'St Kitts Nevis', 'Latvia', 'Suriname', 'Uganda',
       'El Salvador', 'Bermuda', 'Kuwait', 'Antigua & Barbuda',
       'Thailand', 'Mauritius', 'Guatemala', 'Liberia', 'Kazakhstan',
       'Niger', 'Mauritania', 'Montserrat', 'Namibia', 'Azerbaijan',
       'Guam', 'Faroe Islands', 'India', 'Nicaragua', 'Barbados',
       'Lebanon', 'Palestine', 'Guyana', 'Sudan', 'St Lucia', 'Ethiopia',
       'Puerto Rico', 'Grenada', 'Jordan', 'Rwanda', 'Qatar',
       'Afghanistan', 'Hong Kong', 'Andorra', 'Malta', 'Belize',
       'South Sudan', 'Indonesia', 'Botswana'], dtype=object)
```

In [0]:

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

Out[0]:

```
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,  32,  33,  34,  35,  36,  37,  38,
        39,  40,  41,  42,  43,  44,  45,  46,  47,  48,  49,  50,  51,
        52,  53,  54,  55,  56,  57,  58,  59,  60,  61,  62,  63,  64,
        65,  66,  67,  68,  69,  70,  71,  72,  73,  74,  75,  76,  77,
        78,  79,  80,  81,  82,  83,  84,  85,  86,  87,  88,  89,  90,
        91,  92,  93,  94,  95,  96,  97,  98,  99, 100, 101, 102, 103,
       104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116,
       117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129,
       130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142,
       143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155,
       156, 157, 158, 159, 160, 161, 162, 163])
```

In [0]:

```
le.inverse_transform([x for x in range(164)])
```

Out[0]:

```
array(['Afghanistan', 'Albania', 'Algeria', 'Andorra', 'Angola',
       'Antigua & Barbuda', 'Argentina', 'Armenia', 'Australia',
       'Austria', 'Azerbaijan', 'Barbados', 'Belarus', 'Belgium',
       'Belize', 'Benin', 'Bermuda', 'Bolivia', 'Bosnia Herzegovina',
       'Botswana', 'Brazil', 'Bulgaria', 'Burkina Faso', 'Burundi',
       'Cameroon', 'Canada', 'Cape Verde', 'Central African Rep.', 'Chad',
       'Chile', 'China PR', 'Colombia', 'Comoros', 'Congo', 'Costa Rica',
       'Croatia', 'Cuba', 'Curacao', 'Cyprus', 'Czech Republic',
       'DR Congo', 'Denmark', 'Dominican Republic', 'Ecuador', 'Egypt',
       'El Salvador', 'England', 'Equatorial Guinea', 'Eritrea',
       'Estonia', 'Ethiopia', 'FYR Macedonia', 'Faroe Islands', 'Fiji',
       'Finland', 'France', 'Gabon', 'Gambia', 'Georgia', 'Germany',
       'Ghana', 'Greece', 'Grenada', 'Guam', 'Guatemala', 'Guinea',
       'Guinea Bissau', 'Guyana', 'Haiti', 'Honduras', 'Hong Kong',
       'Hungary', 'Iceland', 'India', 'Indonesia', 'Iran', 'Iraq',
       'Israel', 'Italy', 'Ivory Coast', 'Jamaica', 'Japan', 'Jordan',
       'Kazakhstan', 'Kenya', 'Korea DPR', 'Korea Republic', 'Kosovo',
       'Kuwait', 'Latvia', 'Lebanon', 'Liberia', 'Libya', 'Liechtenstein',
       'Lithuania', 'Luxembourg', 'Madagascar', 'Mali', 'Malta',
       'Mauritania', 'Mauritius', 'Mexico', 'Moldova', 'Montenegro',
       'Montserrat', 'Morocco', 'Mozambique', 'Namibia', 'Netherlands',
       'New Caledonia', 'New Zealand', 'Nicaragua', 'Niger', 'Nigeria',
       'Northern Ireland', 'Norway', 'Oman', 'Palestine', 'Panama',
       'Paraguay', 'Peru', 'Philippines', 'Poland', 'Portugal',
       'Puerto Rico', 'Qatar', 'Republic of Ireland', 'Romania', 'Russia',
       'Rwanda', 'Saudi Arabia', 'Scotland', 'Senegal', 'Serbia',
       'Sierra Leone', 'Slovakia', 'Slovenia', 'South Africa',
       'South Sudan', 'Spain', 'St Kitts Nevis', 'St Lucia', 'Sudan',
       'Suriname', 'Sweden', 'Switzerland', 'Syria',
       'São Tomé & Príncipe', 'Tanzania', 'Thailand', 'Togo',
       'Trinidad & Tobago', 'Tunisia', 'Turkey', 'Uganda', 'Ukraine',
       'United Arab Emirates', 'United States', 'Uruguay', 'Uzbekistan',
       'Venezuela', 'Wales', 'Zambia', 'Zimbabwe'], dtype=object)
```

One-hot encording

In [0]:

```
ohe = OneHotEncoder()
cat_enc_ohe = ohe.fit_transform(cat_enc[['c1']])
```

In [0]:

```
cat_enc.shape
```

Out[0]:

```
(18207, 1)
```

In [0]:

```
cat_enc_ohe.shape
```

Out[0]:

```
(18207, 164)
```

In [0]:

```
cat_enc_ohe
```

Out[0]:

```
<18207x164 sparse matrix of type '<class 'numpy.float64'>'
        with 18207 stored elements in Compressed Sparse Row format>
```

In [0]:

```
cat_enc_ohe.todense()[0:163]
```

Out[0]:

```
matrix([[0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.],
        ...,
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.],
        [0., 0., 0., ..., 0., 0., 0.]])
```

In [0]:

```
cat_enc.head(100)
```

Out[0]:

	c1
0	Argentina
1	Portugal
2	Brazil
3	Spain
4	Belgium
5	Belgium
6	Croatia
7	Uruguay
8	Spain
9	Slovenia
10	Poland
11	Germany
12	Uruguay
13	Spain
14	France
15	Argentina
16	England
17	France
18	Germany
19	Belgium
20	Spain
21	Uruguay
22	Germany
23	Argentina
24	Italy
25	France
26	Egypt
27	Brazil
28	Colombia
29	Italy
...	...
70	Italy
71	Belgium
72	Bosnia Herzegovina
73	Morocco
74	Germany
75	Brazil

76	Spain
77	Slovakia
78	Serbia
79	Spain
80	France
81	Brazil
82	Germany
83	Spain
84	Algeria
85	Austria
86	Spain
87	France
88	Greece
89	Argentina
90	Spain
91	Brazil
92	Poland
93	Chile
94	Algeria
95	Germany
96	Chile
97	Croatia
98	Bosnia Herzegovina
99	Germany

100 rows × 1 columns