1. Рубежный контроль №1

Смирнов А. И., группа ИУ5-22М. Вариант №3, набор данных №2.

1.1. Задание

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?

1.2. Решение

1.2.1. Загрузка и предобработка данных

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
```

[2]:
```
data = pd.read_csv("dc-wikia-data.csv")
```

[3]:
```
data.dtypes
```

[3]:
```
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
```

[4]:
```
data.head()
```

[4]:

	page_id	name	urlslug \
0	1422	Batman (Bruce Wayne)	\/wiki\/Batman_(Bruce_Wayne)
1	23387	Superman (Clark Kent)	\/wiki\/Superman_(Clark_Kent)
2	1458	Green Lantern (Hal Jordan)	\/wiki\/Green_Lantern_(Hal_Jordan)
3	1659	James Gordon (New Earth)	\/wiki\/James_Gordon_(New_Earth)
4	1576	Richard Grayson (New Earth)	\/wiki\/Richard_Grayson_(New_Earth)

	ID	ALIGN	EYE	HAIR	SEX \
0	Secret Identity	Good Characters	Blue Eyes	Black Hair	Male Characters
1	Secret Identity	Good Characters	Blue Eyes	Black Hair	Male Characters
2	Secret Identity	Good Characters	Brown Eyes	Brown Hair	Male Characters

```
3  Public Identity  Good Characters  Brown Eyes  White Hair  Male Characters
4  Secret Identity  Good Characters   Blue Eyes  Black Hair  Male Characters

   GSM            ALIVE  APPEARANCES FIRST APPEARANCE    YEAR
0  NaN  Living Characters       3093.0       1939, May  1939.0
1  NaN  Living Characters       2496.0    1986, October  1986.0
2  NaN  Living Characters       1565.0    1959, October  1959.0
3  NaN  Living Characters       1316.0   1987, February  1987.0
4  NaN  Living Characters       1237.0      1940, April  1940.0
```

[5]:
```
data.shape
```

[5]:
```
(6896, 13)
```

[6]:
```
data.isnull().sum()
```

[6]:
```
page_id              0
name                 0
urlslug              0
ID                2013
ALIGN              601
EYE               3628
HAIR              2274
SEX                125
GSM               6832
ALIVE                3
APPEARANCES        355
FIRST APPEARANCE    69
YEAR                69
dtype: int64
```

[7]:
```
d = data[["name", "SEX", "APPEARANCES"]]
d = d.dropna(axis=0, how="any")
```

[8]:
```
d.head()
```

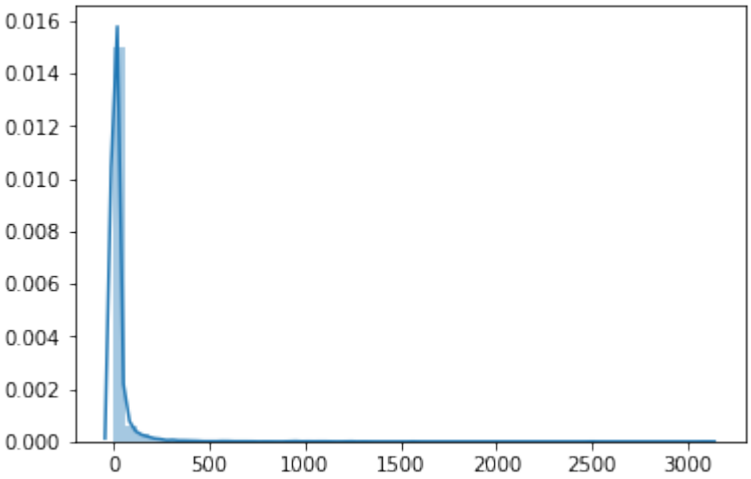
[8]:
```
                     name            SEX  APPEARANCES
0       Batman (Bruce Wayne)  Male Characters       3093.0
1       Superman (Clark Kent)  Male Characters       2496.0
2   Green Lantern (Hal Jordan)  Male Characters       1565.0
3    James Gordon (New Earth)  Male Characters       1316.0
4  Richard Grayson (New Earth)  Male Characters       1237.0
```

[9]:
```
d.shape
```

[9]:
```
(6427, 3)
```

1.2.2. Масштабирование данных

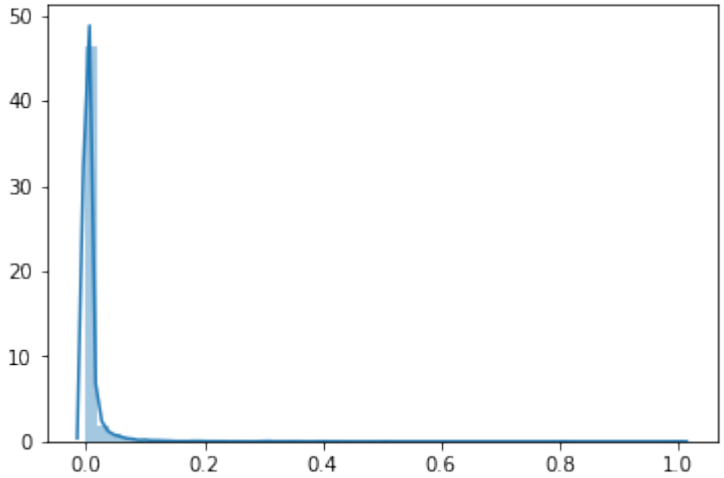
[10]:
```
sns.distplot(d[["APPEARANCES"]]);
```



[11]:
```
from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler()
sc_data = sc.fit_transform(d[["APPEARANCES"]])

sns.distplot(sc_data)
```

[11]: <matplotlib.axes._subplots.AxesSubplot at 0x169b1e72dd8>



[12]:
```
d["APPEARANCES_SCALED"] = sc_data
```

1.2.3. Преобразование категориальных признаков

[13]:
```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

Label encoding

[14]:
```
le = LabelEncoder()
sex_le = le.fit_transform(d["SEX"])
```

[15]:
```
np.unique(sex_le)
```

[15]:
```
array([0, 1, 2, 3])
```

[16]:
```
le.inverse_transform(np.unique(sex_le))
```

[16]:
```
array(['Female Characters', 'Genderless Characters', 'Male Characters',
       'Transgender Characters'], dtype=object)
```

[17]:
```
d["SEX_INDEX"] = sex_le
```

One Hot encoding

[18]:
```
ohe = OneHotEncoder()
sex_ohe = ohe.fit_transform(d[["SEX"]])
```

[19]:
```
sex_ohe.todense()[0:10]
```

[19]:
```
matrix([[0., 0., 1., 0.],
        [0., 0., 1., 0.],
        [0., 0., 1., 0.],
        [0., 0., 1., 0.],
        [0., 0., 1., 0.],
        [1., 0., 0., 0.],
        [0., 0., 1., 0.],
        [0., 0., 1., 0.],
        [1., 0., 0., 0.],
        [0., 0., 1., 0.]])
```

[20]:
```
d["SEX"].head(10)
```

[20]:
```
0      Male Characters
1      Male Characters
2      Male Characters
3      Male Characters
4      Male Characters
5    Female Characters
6      Male Characters
7      Male Characters
8    Female Characters
9      Male Characters
Name: SEX, dtype: object
```

[21]:
```
ohe_names = ohe.get_feature_names()
ohe_names
```

[21]:
```
array(['x0_Female Characters', 'x0_Genderless Characters',
       'x0_Male Characters', 'x0_Transgender Characters'], dtype=object)
```

[22]:
```
for idx, name in enumerate(ohe_names):
    d[name] = sex_ohe[:, idx].todense()
```

1.2.4. Получившийся набор данных

[23]:
```
d.head(10)
```

[23]:

	name	SEX	APPEARANCES	APPEARANCES_SCALED	SEX_INDEX	x0_Female Characters	x0_Genderless Characters	x0_Male Characters	x0_Transgender Characters
0	Batman (Bruce Wayne)	Male Characters	3093.0	1.000000	2	0.0	0.0	1.0	0.0
1	Superman (Clark Kent)	Male Characters	2496.0	0.806921	2	0.0	0.0	1.0	0.0
2	Green Lantern (Hal Jordan)	Male Characters	1565.0	0.505821	2	0.0	0.0	1.0	0.0
3	James Gordon (New Earth)	Male Characters	1316.0	0.425291	2	0.0	0.0	1.0	0.0
4	Richard Grayson (New Earth)	Male Characters	1237.0	0.399741	2	0.0	0.0	1.0	0.0
5	Wonder Woman (Diana Prince)	Female Characters	1231.0	0.397801	0	1.0	0.0	0.0	0.0
6	Aquaman (Arthur Curry)	Male Characters	1121.0	0.362225	2	0.0	0.0	1.0	0.0
7	Timothy Drake (New Earth)	Male Characters	1095.0	0.353816	2	0.0	0.0	1.0	0.0
8	Dinah Laurel Lance (New Earth)	Female Characters	1075.0	0.347348	0	1.0	0.0	0.0	0.0
9	Flash (Barry Allen)	Male Characters	1028.0	0.332147	2	0.0	0.0	1.0	0.0