Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №3
по дисциплине

«Методы машинного обучения»
на тему

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

Выполнил:
студент группы ИУ5-21М
Исмаил Ахмад

Москва — 2020 г.

1. Цель лабораторной работы

Изучить способы предварительной обработки данных для дальнейшего формирования моделей

2. Задание

Требуется [1]:

1. Выбрать набор данных (датасет), содержащий категориальные признаки и пропуски в данных. Для выполнения следующих пунктов можно использовать несколько различных наборов данных.
2. Для выбранного датасета (датасетов) на основе материалов лекции решить следующие задачи:

• обработку пропусков в данных;
• кодирование категориальных признаков;
• масштабирование данных.

3. Ход выполнения работы

Подключим все необходимые библиотеки и настроим отображение графиков [2,3]:

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import sklearn.impute
import sklearn.preprocessing
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19: FutureWarning: pandas.util.testin
g is deprecated. Use the functions in the public API at pandas.testing instead.
  import pandas.util.testing as tm
```

 # Enable inline plots

In [0]:
```
%matplotlib inline
```

 # Set plot style

In [0]:
```
sns.set(style="ticks")
```

In [0]:
```
from IPython.display import set_matplotlib_formats
set_matplotlib_formats("retina")
```

тобы в дальнейшем текст в отчёте влезал на А4 [4]:

In [0]:
```
pd.set_option("display.width", 70)
```

Для выполнения данной лабораторной работы возьмём набор данных по приложениям в US counties COVID 19 dataset

In [0]:
```
data = pd.read_csv("/content/sample_data/us-counties.csv")
```

In [7]:
```
data.head()
```

Out[7]:

	date	county	state	fips	cases	deaths
0	2020-01-21	Snohomish	Washington	53061.0	1	0
1	2020-01-22	Snohomish	Washington	53061.0	1	0
2	2020-01-23	Snohomish	Washington	53061.0	1	0
3	2020-01-24	Cook	Illinois	17031.0	1	0
4	2020-01-24	Snohomish	Washington	53061.0	1	0

In [8]:
```
data.dtypes
```

Out[8]:
```
date       object
county     object
state      object
fips      float64
cases       int64
deaths      int64
dtype: object
```

In [9]:
```
data.shape
```

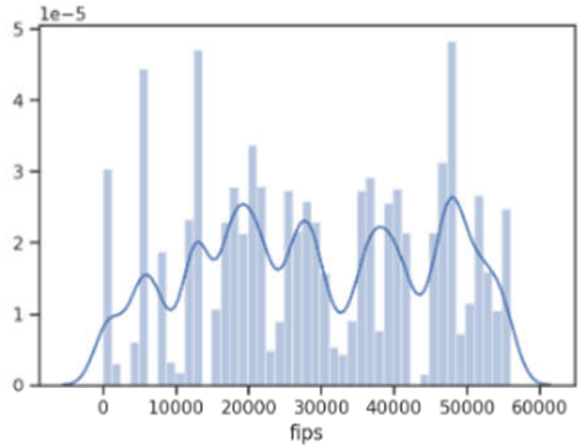
Out[9]:
```
(64707, 6)
```

Обработка пропусков в данных

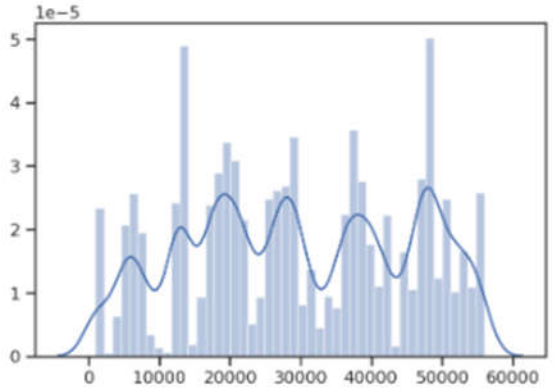
In [10]:
```
data.isnull().sum()
```

Out[10]:
```
date        0
county      0
state       0
fips      838
cases       0
deaths      0
dtype: int64
```

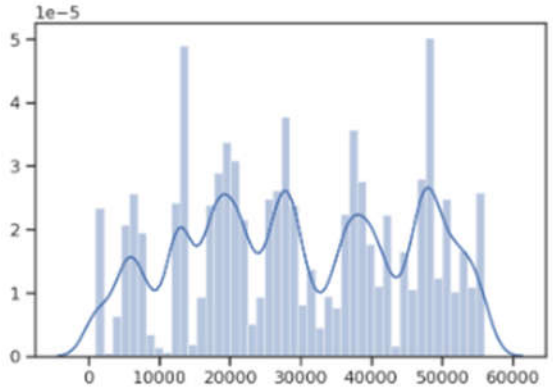
In [11]:
```
sns.distplot(data["fips"].fillna(0));
```



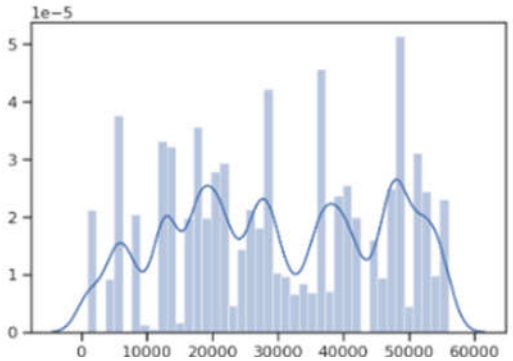
In [12]:
```
mean_imp = sklearn.impute.SimpleImputer(strategy="mean")
mean_rat = mean_imp.fit_transform(data[["fips"]])
sns.distplot(mean_rat);
```



In [13]:
```
med_imp = sklearn.impute.SimpleImputer(strategy="median")
med_rat = med_imp.fit_transform(data[["fips"]])
sns.distplot(med_rat);
```



In [14]:
```
freq_imp = sklearn.impute.SimpleImputer(strategy="most_frequent")
freq_rat = freq_imp.fit_transform(data[["fips"]])
sns.distplot(freq_rat);
```



Кодирование категориальных признаков

In [15]:
```
types = data["state"].dropna().astype(str)
types.value_counts()
```

Out[15]:

Texas	3877
Georgia	3814
Virginia	2706
Indiana	2236
North Carolina	2226
Tennessee	2175
Mississippi	2080
Kentucky	2058
Ohio	2052
California	1990
Florida	1896
Missouri	1895
Illinois	1869
Michigan	1860
New York	1780
Pennsylvania	1724
Iowa	1711
Louisiana	1696
Arkansas	1626
Alabama	1597
Minnesota	1590
Colorado	1531
Wisconsin	1516
Oklahoma	1361
Washington	1287
South Carolina	1266
Kansas	1258
Oregon	850
West Virginia	840
Nebraska	839
South Dakota	814
New Jersey	763
Idaho	723
Maryland	718
Montana	627
North Dakota	613
Utah	593
New Mexico	591
Massachusetts	559
Arizona	490
Wyoming	474
Maine	420
Vermont	403
New Hampshire	316
Nevada	283
Connecticut	278
Alaska	242
Hawaii	165
Rhode Island	162
Delaware	104
District of Columbia	41
Puerto Rico	35
Virgin Islands	34
Guam	33
Northern Mariana Islands	20

Name: state, dtype: int64

In [16]:
```
le = sklearn.preprocessing.LabelEncoder()
type_le = le.fit_transform(types)
print(np.unique(type_le))
le.inverse_transform(np.unique(type_le))
```

```
[ 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]
```

Out[16]:
```
array(['Alabama', 'Alaska', 'Arizona', 'Arkansas', 'California',
       'Colorado', 'Connecticut', 'Delaware', 'District of Columbia',
       'Florida', 'Georgia', 'Guam', 'Hawaii', 'Idaho', 'Illinois',
       'Indiana', 'Iowa', 'Kansas', 'Kentucky', 'Louisiana', 'Maine',
       'Maryland', 'Massachusetts', 'Michigan', 'Minnesota',
       'Mississippi', 'Missouri', 'Montana', 'Nebraska', 'Nevada',
       'New Hampshire', 'New Jersey', 'New Mexico', 'New York',
       'North Carolina', 'North Dakota', 'Northern Mariana Islands',
       'Ohio', 'Oklahoma', 'Oregon', 'Pennsylvania', 'Puerto Rico',
       'Rhode Island', 'South Carolina', 'South Dakota', 'Tennessee',
       'Texas', 'Utah', 'Vermont', 'Virgin Islands', 'Virginia',
       'Washington', 'West Virginia', 'Wisconsin', 'Wyoming'],
      dtype=object)
```

In [17]:
```
type_oh = pd.get_dummies(types)
type_oh.head()
```

Out[17]:

	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut	Delaware	District of Columbia	Florida	Georgia	Guam	
0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	

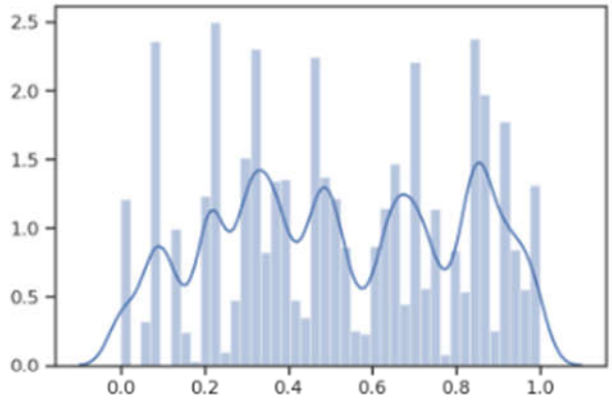
In [18]:
```
type_oh[type_oh["Arizona"] == 1].head()
```

Out[18]:

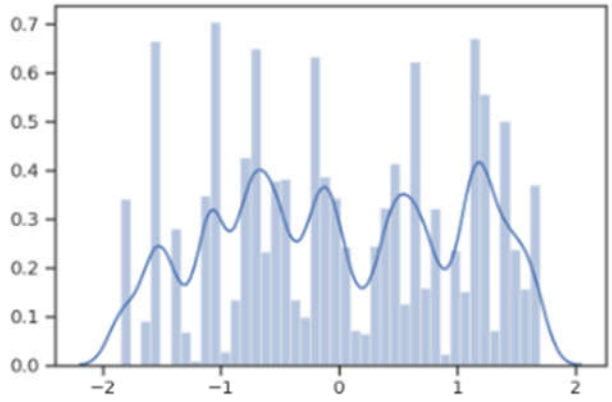
	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut	Delaware	District of Columbia	Florida	Georgia	Guam
8	0	0	1	0	0	0	0	0	0	0	0	0
13	0	0	1	0	0	0	0	0	0	0	0	0
18	0	0	1	0	0	0	0	0	0	0	0	0
23	0	0	1	0	0	0	0	0	0	0	0	0
28	0	0	1	0	0	0	0	0	0	0	0	0

Масштабирование данных

In [19]:
```
mm = sklearn.preprocessing.MinMaxScaler()
sns.distplot(mm.fit_transform(data[["fips"]]));
```



In [20]:
```
ss = sklearn.preprocessing.StandardScaler()
sns.distplot(ss.fit_transform(data[["fips"]]));
```



Список литературы

[1] Гапанюк Ю. Е. Лабораторная работа «Обработка пропусков в данных, кодирование категориальных признаков, масштабирование данных» [Электронный ресурс] // GitHub. — 2019. — Режим доступа: https://github.com/ugapanyuk/ml_course/wiki/LAB_MISSING (дата обращения: 05.04.2019).

[2] Team The IPython Development. IPython 7.3.0 Documentation [Electronic resource] // Read the Docs. — 2019. — Access mode: https://ipython.readthedocs.io/en/stable/ (online; accessed: 20.02.2019).

[3] Waskom M. seaborn 0.9.0 documentation [Electronic resource] // PyData. — 2018. — Access mode: https://seaborn.pydata.org/ (online; accessed: 20.02.2019).

[4] pandas 0.24.1 documentation [Electronic resource] // PyData. — 2019. — Access mode: http://pandas.pydata.org/pandas-docs/stable/ (online; accessed: 20.02.2019).

[5] Gupta L. Google Play Store Apps [Electronic resource] // Kaggle. — 2019. — Access mode: https://www.kaggle.com/lava18/google-play-store-apps (online; accessed: 05.04.2019)