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 ИУ5 «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Отчет по рубежному контролю №1
Вариант 16

Выполнила:

студент группы ИУ5-61Б

Рогозин Д.Р.

18.04.2021

Проверил:

преподаватель каф. ИУ5

Гапанюк Ю.Е.

Москва, 2021 г.

Задание:

Для заданного набора данных проведите обработку пропусков в данных для одного категориального и одного количественного признака. Какие способы обработки пропусков в данных для категориальных и количественных признаков Вы использовали? Какие признаки Вы будете использовать для дальнейшего построения моделей машинногообучения и почему?

Для студентов групп ИУ5-61Б, ИУ5Ц-81Б - для пары произвольных колонок данных построить график "Диаграмма рассеяния".

1. Набор данных: https://www.kaggle.com/lava18/google-play-store-apps

Ответы на вопросы:

Для обработки пропусков данных для количественного признака Rating я использовал импьютацию различными показателями центра распределения 'mean', 'median', most_frequent' (среднее значение, медиана, мода) с помощью класса SimpleImputer библиотеки scikit-learn.
Для обработки пропусков данных для категориального признака Rating я также использовал класс SimpleImputer со стратегиями 'most_frequent' или 'constant' (мода иконстанта).

Текст программы и экранные формы с примерами выполнения программы (ячейкиноутбука):

ИУ5-61Б Рогозин Д.Р. РК1 ТМО

Вариант 16

In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.datasets import *
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [2]:
```
# Загрузка данных
data = pd.read_csv('googleplaystore.csv')
```

In [3]:
```
# Обзор датасета
data.head()
```

Out[3]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [4]:
```
data.shape
```

Out[4]:
```
(10841, 13)
```

In [5]:
```
data.columns
```

Out[5]:
```
Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
       'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
       'Android Ver'],
      dtype='object')
```

In [6]:
```
data.dtypes
```

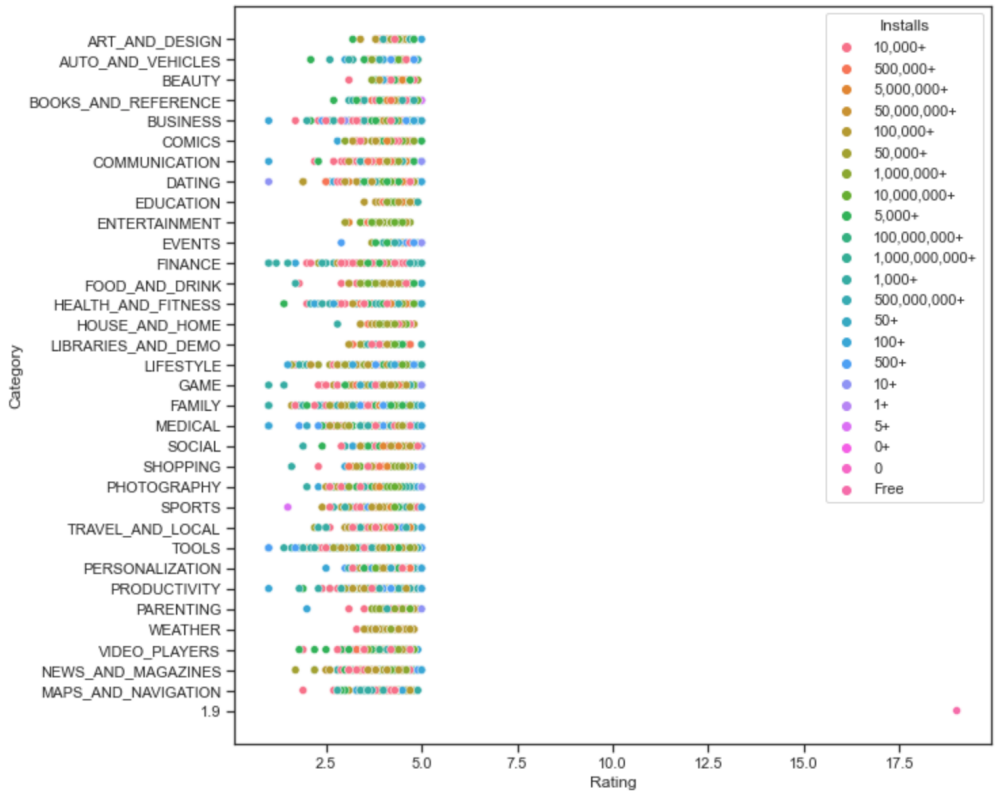
Out[6]:
```
App                object
Category           object
Rating            float64
Reviews            object
Size               object
Installs           object
Type               object
Price              object
Content Rating     object
Genres             object
Last Updated       object
Current Ver        object
Android Ver        object
dtype: object
```

In [7]:
```
# Проверка наличия пустых значений
# Цикл по колонкам датасета
for col in data.columns:
    # Количество пустых значений - все значения заполнены
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
```

```
App - 0
Category - 0
Rating - 1474
Reviews - 0
Size - 0
Installs - 0
Type - 1
Price - 0
Content Rating - 1
Genres - 0
Last Updated - 0
Current Ver - 8
Android Ver - 3
```

In [8]:
```
# Диаграмма рассеивания
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='Rating', y='Category', data=data,hue = 'Installs')
```

Out[8]:
```
<AxesSubplot:xlabel='Rating', ylabel='Category'>
```



Удаление или заполнение нулями

In [10]:
```
# Удаление колонок, содержащих пустые значения
# В данном случае такое удаление колонок некорректно, так как рейтинг одна из самых важных метрик
data_new_1 = data.dropna(axis=1, how='any')
(data.shape, data_new_1.shape)
```

Out[10]: ((10841, 13), (10841, 8))

In [11]:
```
# Удаление строк, содержащих пустые значения
data_new_2 = data.dropna(axis=0, how='any')
(data.shape, data_new_2.shape)
```

Out[11]: ((10841, 13), (9360, 13))

In [12]:
```
data.head()
```

Out[12]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

In [13]:
```
# Заполнение всех пропущенных значений нулями
# Для данного датасета способ не подходит, так как содержатся категориальные признаки с пропусками
data_new_3 = data.fillna(0)
data_new_3.head()
```

Out[13]:

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

"Внедрение значений" - импьютация Обработка пропусков для количественного признака Rating

In [14]:
```
# Выберем числовые колонки с пропущенными значениями
# Цикл по колонкам датасета
num_cols = []
for col in data.columns:
    # Количество пустых значений
    temp_null_count = data[data[col].isnull()].shape[0]
    dt = str(data[col].dtype)
    if temp_null_count>0 and (dt=='float64'):
        num_cols.append(col)
        temp_perc = round((temp_null_count / total_count) * 100.0, 2)
        print('Колонка {}. Тип данных {}. Количество пустых значений {}, {}%.'.format(col, dt, temp_null_count, temp
```

```
Колонка Rating. Тип данных float64. Количество пустых значений 1474, 13.6%.
```

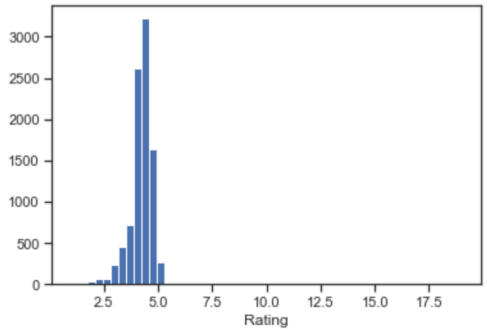
In [15]:
```
# Фильтр по колонкам с пропущенными значениями
data_num = data[num_cols]
data_num
```

Out[15]:

	Rating
0	4.1
1	3.9
2	4.7
3	4.5
4	4.3
...	...
10836	4.5
10837	5.0
10838	NaN
10839	4.5
10840	4.5

10841 rows × 1 columns

In [16]:
```
# Гистограмма по признакам
for col in data_num:
    plt.hist(data[col], 50)
    plt.xlabel(col)
    plt.show()
```



In [17]:
```
data_num_Rating = data_num[['Rating']]
data_num_Rating.head()
```

Out[17]:

	Rating
0	4.1
1	3.9
2	4.7
3	4.5
4	4.3

In [18]:
```
from sklearn.impute import SimpleImputer
from sklearn.impute import MissingIndicator
```

In [19]:
```
# Фильтр для проверки заполнения пустых значений
indicator = MissingIndicator()
mask_missing_values_only = indicator.fit_transform(data_num_Rating)
mask_missing_values_only
```

Out[19]:
```
array([[False],
       [False],
       [False],
       ...,
       [ True],
       [False],
       [False]])
```

In [20]:
```
# Импьютация различными показателями центра распределения с помощью класса SimpleImputer
strategies=['mean', 'median', 'most_frequent']
```

In [21]:
```
def test_num_impute(strategy_param):
    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(data_num_Rating)
    return data_num_imp[mask_missing_values_only]
```

In [22]:
```
# Среднее значение
strategies[0], test_num_impute(strategies[0])
```

Out[22]:
```
('mean',
 array([4.19333832, 4.19333832, 4.19333832, ..., 4.19333832, 4.19333832,
        4.19333832]))
```

In [23]:
```
# Медиана
strategies[1], test_num_impute(strategies[1])
```

Out[23]:
```
('median', array([4.3, 4.3, 4.3, ..., 4.3, 4.3, 4.3]))
```

In [24]:
```
# Мода
strategies[2], test_num_impute(strategies[2])
```

Out[24]:
```
('most_frequent', array([4.4, 4.4, 4.4, ..., 4.4, 4.4, 4.4]))
```

Обработка пропусков для категориального признака Current Ver

In [25]:
```
# Выбор категориальных колонок с пропущенными значениями
# Цикл по колонкам датасета
cat_cols = []
for col in data.columns:
    # Количество пустых значений
    temp_null_count = data[data[col].isnull()].shape[0]
    dt = str(data[col].dtype)
    if temp_null_count>0 and (dt=='object'):
        cat_cols.append(col)
        temp_perc = round((temp_null_count / total_count) * 100.0, 2)
        print('Колонка {}. Тип данных {}. Количество пустых значений {}, {}%.'.format(col, dt, temp_null_count, temp
```

```
Колонка Type. Тип данных object. Количество пустых значений 1, 0.01%.
Колонка Content Rating. Тип данных object. Количество пустых значений 1, 0.01%.
Колонка Current Ver. Тип данных object. Количество пустых значений 8, 0.07%.
Колонка Android Ver. Тип данных object. Количество пустых значений 3, 0.03%.
```

In [26]:
```
# Импьютация с помощью класса SimpleImputer со стратегиями "most_frequent" или "constant".
cat_temp_data = data[['Current Ver']]
cat_temp_data.head()
```

Out[26]:

	Current Ver
0	1.0.0
1	2.0.0
2	1.2.4
3	Varies with device
4	1.1

In [27]:
```
cat_temp_data['Current Ver'].unique()
```

Out[27]:
```
array(['1.0.0', '2.0.0', '1.2.4', ..., '1.0.612928', '0.3.4', '2.0.148.0'],
      dtype=object)
```

In [28]:
```
cat_temp_data[cat_temp_data['Current Ver'].isnull()].shape
```

Out[28]:
```
(8, 1)
```

In [29]:
```
# Импьютация наиболее частыми значениями (мода)
imp2 = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
data_imp2 = imp2.fit_transform(cat_temp_data)
data_imp2
```

Out[29]:
```
array([['1.0.0'],
       ['2.0.0'],
       ['1.2.4'],
       ...,
       ['1.0'],
       ['Varies with device'],
       ['Varies with device']], dtype=object)
```

In [30]:
```
# Пустые значения отсутствуют
np.unique(data_imp2)
```

Out[30]:
```
array(['0.0.0.2', '0.0.1', '0.0.10', ..., 'v8.0.1.8.0629.1', 'v8[1.0.10]',
       'version 0.994'], dtype=object)
```

In [31]:
```
# Импьютация константой
imp3 = SimpleImputer(missing_values=np.nan, strategy='constant', fill_value='NULL')
data_imp3 = imp3.fit_transform(cat_temp_data)
data_imp3
```

Out[31]:
```
array([['1.0.0'],
       ['2.0.0'],
       ['1.2.4'],
       ...,
       ['1.0'],
       ['Varies with device'],
       ['Varies with device']], dtype=object)
```

In [32]:
```
np.unique(data_imp3)
```

Out[32]:
```
array(['0.0.0.2', '0.0.1', '0.0.10', ..., 'v8.0.1.8.0629.1', 'v8[1.0.10]',
       'version 0.994'], dtype=object)
```

In [33]:
```
data_imp3[data_imp3=='NULL'].size
```