Рубежный контроль №1¶

Карягин А.Д., группа ИУ5-62Б, вариант 8 (задача №1, набор данных №8)

Для заданного набора данных проведите корреляционный анализ. В случае наличия пропусков в данных удалите строки или колонки, содержащие пропуски. Сделайте выводы о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

Требование для студентов группы ИУ5-62Б - для произвольной колонки данных построить гистограмму.

In [ ]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import re
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
%matplotlib inline
sns.set(style="ticks")
```

In [125]:

```
data = pd.read_csv('data/google-play-store-apps/googleplaystore.csv', sep=",")
```

In [126]:

```
data.head()
```

Out[126]:

	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

Удаляем Категориальные признаки которые не будем кодировать¶

In [127]:

```
data.drop(['App', 'Category', 'Price', 'Genres', 'Last Updated', 'Current Ver', 'Android Ver'], axis='columns', inplace=True)
```

In [128]:

```
data.head()
```

Out[128]:

	Rating	Reviews	Size	Installs	Type	Content Rating
0	4.1	159	19M	10,000+	Free	Everyone
1	3.9	967	14M	500,000+	Free	Everyone
2	4.7	87510	8.7M	5,000,000+	Free	Everyone
3	4.5	215644	25M	50,000,000+	Free	Teen
4	4.3	967	2.8M	100,000+	Free	Everyone

In [ ]:

In [129]:

```
data.isnull().sum()
```

Out[129]:

```
Rating            1474
Reviews              0
Size                 0
Installs             0
Type                 1
Content Rating       1
dtype: int64
```

In [130]:

```
data.shape
```

Out[130]:

```
(10841, 6)
```

Так как Рейтинг может быть целевым признаком удалять этот столбец мы не будем. Пропуски в остальных столбцах малочисленны, так что удалим все строки с пропусками.

In [131]:

```
data.dropna(axis = 0, inplace= True)
```

In [132]:

```
data.isnull().sum()
```

Out[132]:

```
Rating            0
Reviews           0
Size              0
Installs          0
Type              0
Content Rating    0
dtype: int64
```

In [133]:

```
data.corr()
```

Out[133]:

	Rating
Rating	1.0

Нужно преобразовать категориальные признаки в числовые

In [134]:

```
install_uniq = data['Installs'].unique()
install_dict = {uniq : re.sub(r"[+,]", "", uniq) for uniq in install_uniq}
data = data.replace({"Installs": install_dict})
data['Installs'] = data['Installs'].astype('int32')
```

In [135]:

```
data['Installs'].unique()
```

Out[135]:

```
array([     10000,     500000,    5000000,   50000000,     100000,
         1000000,    1000000,   10000000,       5000, 100000000,
       1000000000,       1000,     500000000,        100,        500,
               10,          5,          50,          1], dtype=int32)
```

In [136]:

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

Out[136]:

	Content
Rating	4.1
Reviews	159
Size	19M
Installs	10000
Type	Free
Content Rating	Everyone

In [137]:

```
le = LabelEncoder()
le.fit(data['Content Rating'])
data['Content_rating_le'] = le.transform(data['Content Rating'])
data
```

Out[137]:

	Rating	Reviews	Size	Installs	Type	Content Rating	Content_rating_le
0	4.1	159	19M	10000	Free	Everyone	1
1	3.9	967	14M	500000	Free	Everyone	1
2	4.7	87510	8.7M	5000000	Free	Everyone	1
3	4.5	215644	25M	50000000	Free	Teen	4
4	4.3	967	2.8M	100000	Free	Everyone	1
...	...	...	...	...	...	...	...
10834	4.0	7	2.6M	500	Free	Everyone	1
10836	4.5	38	53M	5000	Free	Everyone	1
10837	5.0	4	3.6M	100	Free	Everyone	1
10839	4.5	114	Varies with device	1000	Free	Mature 17+	3
10840	4.5	398307	19M	10000000	Free	Everyone	1

9366 rows × 7 columns

In [138]:

```
le = LabelEncoder()
le.fit(data['Type'])
data['Type_le'] = le.transform(data['Type'])
data
```

Out[138]:

	Rating	Reviews	Size	Installs	Type	Content Rating	Content_rating_le	Type_le
0	4.1	159	19M	10000	Free	Everyone	1	0
1	3.9	967	14M	500000	Free	Everyone	1	0
2	4.7	87510	8.7M	5000000	Free	Everyone	1	0
3	4.5	215644	25M	50000000	Free	Teen	4	0
4	4.3	967	2.8M	100000	Free	Everyone	1	0
...	...	...	...	...	...	...	...	...
10834	4.0	7	2.6M	500	Free	Everyone	1	0
10836	4.5	38	53M	5000	Free	Everyone	1	0
10837	5.0	4	3.6M	100	Free	Everyone	1	0
10839	4.5	114	Varies with device	1000	Free	Mature 17+	3	0
10840	4.5	398307	19M	10000000	Free	Everyone	1	0

9366 rows × 8 columns

In [122]:

```
data.drop(['Installs'], axis='columns', inplace=True)
```

Корреляционный анализ¶

In [140]:

```
data.corr()
```

Out[140]:

	Rating	Installs	Content_rating_le	Type_le
Rating	1.000000	0.051355	0.019868	0.039581
Installs	0.051355	1.000000	0.053359	-0.053102
Content_rating_le	0.019868	0.053359	1.000000	-0.046892
Type_le	0.039581	-0.053102	-0.046892	1.000000

In [141]:

```
fig, ax = plt.subplots(figsize=(12,12))
sns.heatmap(data.corr(), cmap='YlGnBu', annot=True, fmt='.3f')
```

Out[141]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0c924b9410>
```


In [ ]:

Как видно на корреляционной матрице, признаки коррелируют очень слабо, можно сделать вывод о невозможности построения модели на основе выбранных признаков. Данный набор данных содержит преимущественно категориальные признаки что усложняет анализ ценности каждого из них.

Гистограмма¶

In [145]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.distplot(data['Rating'])
```

Out[145]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0c921c5fd0>
```


In [ ]: