import pandas as pd In [75]: import seaborn as sns import matplotlib.pyplot as plt %matplotlib inline sns.set(style="ticks") data = pd.read_csv('googleplaystore.csv', sep = ",") In [76]: In [77]: data.shape (10841, 13)Out[77]: In [78]: data.dtypes object Out[78]: App object Category float64 Rating object Reviews Size object Installs object Type object Price object Content Rating object Genres object object Last Updated Current Ver object Android Ver object dtype: object Как мы можем видеть, все значения кроме рейтинга являются объектами, хотя не должны (из-за миллионов в виде буквы М, запятых, разделяющих тысячи и плюсов в загрузках), что мы должны преобразовать. In [79]: data.isnull().sum() 0 Out[79]: App 0 Category Rating 1474 Reviews 0 Size 0 Installs Type 1 0 Price Content Rating 1 0 Genres Last Updated 0 Current Ver 8 Android Ver 3 dtype: int64 In [80]: for i in range(len(data['Reviews'])): if data['Reviews'][i][-1] == 'M': print(data['Reviews'][i]) data['Reviews'][i] = str(int(float(data['Reviews'][i][0:-1])) * 1000000) print(data['Reviews'][i]) 3.0M 3000000 <ipython-input-80-f4e9b22d392a>:4: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret urning-a-view-versus-a-copy data['Reviews'][i] = str(int(float(data['Reviews'][i][0:-1])) * 1000000) In [81]: data['Reviews'] = data['Reviews'].astype(str).astype(int) In [82]: for i in range(len(data['Installs'])): if(data['Installs'][i] == 'Free'): data['Installs'][i] = 0 continue data['Installs'][i] = data['Installs'][i].replace(",", "") data['Installs'][i] = data['Installs'][i].replace("+", "") data['Installs'][i] = str(int(float(data['Installs'][i]))) <ipython-input-82-9c6f0fc030bc>:5: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret urning-a-view-versus-a-copy data['Installs'][i] = data['Installs'][i].replace(",", "") <ipython-input-82-9c6f0fc030bc>:6: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret urning-a-view-versus-a-copy data['Installs'][i] = data['Installs'][i].replace("+", "") <ipython-input-82-9c6f0fc030bc>:7: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret urning-a-view-versus-a-copy data['Installs'][i] = str(int(float(data['Installs'][i]))) <ipython-input-82-9c6f0fc030bc>:3: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret urning-a-view-versus-a-copy data['Installs'][i] = 0 data['Installs'] = data['Installs'].astype(str).astype(int) In [83]: data.dtypes In [84]: App object Out[84]: Category object Rating float64 Reviews int64 Size object Installs int64 Type object Price object Content Rating object Genres object Last Updated object Current Ver object Android Ver object dtype: object In [85]: data.head() Out[85]: Content Last **Current Android** Category Rating Reviews Size Installs Type Price App Genres Updated Rating Ver Ver Photo Editor & 4.0.3 Candy January ART_AND_DESIGN 4.1 159 19M 10000 Free 0 Everyone Art & Design 1.0.0 Camera & 7, 2018 and up Grid & ScrapBook Coloring Art & January 4.0.3 ART_AND_DESIGN 500000 Free Design;Pretend 1 3.9 967 14M 2.0.0 book 0 Everyone and up 15, 2018 moana Play U Launcher Lite – **FREE Live** August 4.0.3 ART_AND_DESIGN 4.7 87510 8.7M 5000000 1.2.4 Free 0 Everyone Art & Design and up Cool 1, 2018 Themes, Hide ... Varies Sketch -June 8. 4.2 and 3 Draw & ART_AND_DESIGN 25M 50000000 Art & Design 215644 Teen with 2018 up device Paint Pixel Draw June 20, Number Art & 4.4 and ART_AND_DESIGN 4.3 967 2.8M 100000 Free 0 Everyone 1.1 Art Coloring Design; Creativity 2018 up Book data.describe() In [86]: Out[86]: Rating Reviews Installs count 9367.000000 1.084100e+04 1.084100e+04 4.193338 4.443887e+05 1.546291e+07 mean std 0.537431 2.927728e+06 8.502557e+07 min 25% 4.000000 3.800000e+01 1.000000e+03 50% 4.300000 2.094000e+03 1.000000e+05 **75**% 4.500000 5.479800e+04 5.000000e+06 19.000000 7.815831e+07 1.000000e+09 max Как мы видим теперь основные значения, а именно ревью и загрузки в правильном виде, а значит можем перейти к корреляционному анализу. fig, ax = plt.subplots(figsize=(10,10)) In [87]: sns.scatterplot(ax=ax, x='Rating', y='Installs', data=data) <AxesSubplot:xlabel='Rating', ylabel='Installs'> Out[87]: 1.0 0.8 0.6 Installs 0.4 0.2 0.0 2.5 7.5 12.5 15.0 5.0 10.0 17.5 Rating In [88]: sns.displot(data['Rating']) Out[88]: <seaborn.axisgrid.FacetGrid at 0x7ff28c6a9490> 1000 800 600 400 200 2.5 10.0 12.5 15.0 17.5 5.0 7.5 Rating In [89]: data.corr() Out[89]: Rating Reviews Installs Rating 1.000000 0.067645 0.048652 0.067645 1.000000 0.643084 Reviews Installs 0.048652 0.643084 1.000000 sns.heatmap(data.corr(), annot = True, fmt = '.6f') In [90]: Out[90]: <AxesSubplot:> - 1.0 1.000000 0.067645 0.048652 Rating - 0.8 - 0.6 0.067645 1.000000 0.643084 Reviews - 0.4 1.000000 0.048652 0.643084 0.2 Installs Rating Reviews Installs Сильнее всего отзывы коррелируют с количеством установок, при этом рейтинг не коррелирует практически ни с чем. sns.jointplot(data=data, x='Rating', y='Category') In [91]: <seaborn.axisgrid.JointGrid at 0x7ff2706cb250> BOOKS AND RE TRAVEL AND 12.5 15.0 17.5 5.0 7.5 10.0 Rating sns.jointplot(data=data, x='Reviews', y='Category') In [92]: Out[92]: <seaborn.axisgrid.JointGrid at 0x7ff26f8a06d0> BOOKS_AND_RE TRAVEL AND Reviews In []: