BoardGameML

May 2, 2020

1 Beating the Game

1.0.1 Krissy Gianforte & Dan Kent

Final Project — w207 Section 3, Summer 2018

1.1 Introduction & Background

The board game industry has grown dramatically in the last ten years, with an estimated growth of 30% coming by 2021. This growth is not simply due to a resurgence of the classic games like Monopoly and Scrabble; rather, the very basis of board games is changing. In recent years, the target audience has shifted from children to adults, and new mechanics, goals, and styles have emerged. New genres such as cooperative games, multiday adventures, and puzzle quests have come about - and more are being designed all the time.
Many of these modern board games are getting their start through crowdfunding (for example, Kickstarter). This presents an interesting economic structure: public opinion really determines which games get the chance to come to market. Given only the basic information shared on a Kickstarter page, board game devotees are deciding which games they will support financially.
This model presents an interesting opportunity not previously available: rather than only seeing the most popular, financed games in big box stores, we have data on *all* size games - even those that failed to win the required funding to actually reach release.
This project uses that data to explore what exactly makes a board game successful. Our analysis focuses on games from the last ten years in an effort to capture the new trends that have emerged. We use various models to tease out the features that bring high player enjoyment ratings. At the end of our analysis, we apply our modeling theory to a set of games from 1980-1990, to characterize the important features of that era and potentially identify a concrete difference between games released in different decades.

1.2 Data Intake

The dataset for this project is taken from the kaggle "board games dataset" page (https://www.kaggle.com/gabrio/board-games-dataset/home). The sqlite database captures data combed from the widely-used board games rating site Board Game Geek (http://www.boardgamegeek.com).
(The sqlite file is submitted along with this notebook. It must be located in the same directory to read the data.)

```
In [1]: # Import packages necessary to read sqlite file

        import pandas as pd
        import sqlite3
        import matplotlib.pyplot as plt
        import numpy as np
        %matplotlib inline
        from scipy import stats
        import seaborn as sns
        sns.set()
        pd.set_option('display.max_columns', 100)

In [2]: #Connect to SQLite DB
        conn = sqlite3.connect("database.sqlite")

In [3]: #Create a cursor Object to allow us to execute SQL queries against the DB
        cur = conn.cursor()

In [4]: #Lets look into the different tables
        cur.execute("SELECT name FROM sqlite_master WHERE type='table';")
        print(cur.fetchall())
```

```
[('BoardGames',), ('bgg.ldaOut.topics',), ('bgg.ldaOut.top.terms',), ('bgg.ldaOut.top.documents
```

For this project, we will pull in only the raw data (the BoardGames table) and develop our own models. The raw data is stored in a dataframe called "df_bg".
(The sqlite database also includes some tables developed as part of the kaggle contest, named "lda_x". We will not use those tables here.)

```
In [5]: #Grab the data from the raw BoardGames table
        df_bg = pd.read_sql_query("select * from BoardGames", conn)
        print("Data loaded.")
```

```
Data loaded.
```

```
In [6]: #Close the connection:
        cur.close()
        conn.close()
```

1.3 Data Setup & Cleaning

The Board Games data frame consists of 81 columns (or features) and 90,400 entries. That's a lot of information!

This project hopes to direct board game designers towards qualities that will make their games more successful. In that spirit, we will focus here on features that are under the direct control of the game designer - play time, number of players, mechanics, theme, etc. Our analysis will *not* include input fields that are based on user ratings, since those are only available after a game is released (and therefore would not be available to the game designer during creation).

The output variable for our models will be "success" as measured by player enjoyment using the BoardGameGeek star rating system (a user-submitted 0-10 star rating). Later in this notebook, we will create multiple interpretations of that outcome varaible, including binary "success"/"failure", binned ratings, and the continuous 0-10 scale.

The sections below
(1) Filter the data to remove irrelevant entries
(2) Create variables where needed / translate categorical variables into binary fields
(3) Create a new dataframe containing only the data of interest to us

1.3.1 Data Filtering

```
In [7]: # print a bit of the table to visualize
        df_bg.head()
```

Out[7]:

	row_names	game.id	game.type	details.description	details.image	details.maxplayers	details.maxplaytime	details.minage	details.minplayers	details.minplaytime	details.name	details.playingtime
0	1	1	boardgame	Die Macher is a game about seven sequential po...	//cf.geekdo-images.com/images/pic159509.jpg	5.0	240.0	14.0	3.0	240.0	Die Macher	240.0
1	2	2	boardgame	Dragonmaster is a trick-taking card game based...	//cf.geekdo-images.com/images/pic184174.jpg	4.0	30.0	12.0	3.0	30.0	Dragonmaster	30.0
2	3	3	boardgame	Part of the Knizia tile-laying trilogy, Samura...	//cf.geekdo-images.com/images/pic3211873.jpg	4.0	60.0	10.0	2.0	30.0	Samurai	60.0
3	4	4	boardgame	When you see the triangular box and the luxuri...	//cf.geekdo-images.com/images/pic285299.jpg	4.0	60.0	12.0	2.0	60.0	Tal der Könige	60.0
4	5	5	boardgame	In Acquire, each player strategically invests ...	//cf.geekdo-images.com/images/pic342163.jpg	6.0	90.0	12.0	3.0	90.0	Acquire	90.0

```
                              details.thumbnail  details.yearpublished  \
0   //cf.geekdo-images.com/images/pic159509_t.jpg                 1986.0
1   //cf.geekdo-images.com/images/pic184174_t.jpg                 1981.0
2  //cf.geekdo-images.com/images/pic3211873_t.jpg                 1998.0
3   //cf.geekdo-images.com/images/pic285299_t.jpg                 1992.0
4   //cf.geekdo-images.com/images/pic342163_t.jpg                 1964.0

    attributes.boardgameartist    attributes.boardgamecategory  \
0          Marcus Gschwendtner  Economic,Negotiation,Political
1                   Bob Pepper               Card Game,Fantasy
2               Franz Vohwinkel      Abstract Strategy,Medieval
3                         None                         Ancient
4  Scott Okumura,Peter Whitley                        Economic

  attributes.boardgamecompilation attributes.boardgamedesigner  \
0                            None           Karl-Heinz Schmiel
1                            None        G. W. "Jerry" D'Arcey
2                            None                Reiner Knizia
3                            None            Christian Beierer
4                            None                  Sid Sackson

  attributes.boardgameexpansion  \
0                          None
1                          None
2                          None
3                          None
4                          None

                        attributes.boardgamefamily  \
0        Country: Germany,Valley Games Classic Line
1                                Animals: Dragons
2 Asian Theme,Country: Japan,Knizia tile-laying ...
3            Country: Egypt,Promotional Board Games
4                             3M Bookshelf Series

  attributes.boardgameimplementation attributes.boardgameintegration  \
0                            None                             None
1            Indulgence,Coup d'etat                             None
2                            None                             None
3                            None                             None
4                            None                             None

                       attributes.boardgamemechanic  \
0 Area Control / Area Influence,Auction/Bidding,...
1                                      Trick-taking
2 Area Control / Area Influence,Hand Management,...
3 Action Point Allowance System,Area Control / A...
```

```
4       Hand Management,Stock Holding,Tile Placement

                      attributes.boardgamepublisher  attributes.total  \
0  Hans im Glück Verlags-GmbH,Moskito Spiele,Vall...                6.0
1                           E.S. Lowe,Milton Bradley                7.0
2  999 Games,ABACUSSPIELE,Astrel Games,Ceilikan J...                6.0
3                                           KOSMOS                5.0
4  3M,Avalon Hill,Avalon Hill (Hasbro),Dujardin,G...                6.0

   stats.average  stats.averageweight  stats.bayesaverage  \
0        7.66508               4.3477             7.29168
1        6.60815               1.9423             5.87150
2        7.44119               2.5085             7.28295
3        6.60675               2.6667             5.76636
4        7.35830               2.5089             7.21895

   stats.family.abstracts.bayesaverage  stats.family.abstracts.pos  \
0                                  NaN                         NaN
1                                  NaN                         NaN
2                                  NaN                         NaN
3                                  NaN                         NaN
4                                  NaN                         NaN

   stats.family.cgs.bayesaverage  stats.family.cgs.pos  \
0                            NaN                   NaN
1                            NaN                   NaN
2                            NaN                   NaN
3                            NaN                   NaN
4                            NaN                   NaN

   stats.family.childrensgames.bayesaverage  stats.family.childrensgames.pos  \
0                                       NaN                              NaN
1                                       NaN                              NaN
2                                       NaN                              NaN
3                                       NaN                              NaN
4                                       NaN                              NaN

   stats.family.familygames.bayesaverage  stats.family.familygames.pos  \
0                                    NaN                           NaN
1                                    NaN                           NaN
2                                    NaN                           NaN
3                                    NaN                           NaN
4                                    NaN                           NaN

   stats.family.partygames.bayesaverage  stats.family.partygames.pos  \
0                                   NaN                          NaN
1                                   NaN                          NaN
2                                   NaN                          NaN
```

```
3                                    NaN                           NaN
4                                    NaN                           NaN

   stats.family.strategygames.bayesaverage  stats.family.strategygames.pos  \
0                                  7.39570                            85.0
1                                  5.91318                          1066.0
2                                  7.30610                           112.0
3                                      NaN                             NaN
4                                  7.21696                           141.0

   stats.family.thematic.bayesaverage  stats.family.thematic.pos  \
0                                 NaN                        NaN
1                                 NaN                        NaN
2                                 NaN                        NaN
3                                 NaN                        NaN
4                                 NaN                        NaN

   stats.family.wargames.bayesaverage  stats.family.wargames.pos  \
0                                 NaN                        NaN
1                                 NaN                        NaN
2                                 NaN                        NaN
3                                 NaN                        NaN
4                                 NaN                        NaN

   stats.median  stats.numcomments  stats.numweights  stats.owned  \
0           0.0             1763.0             719.0       5251.0
1           0.0              273.0              52.0       1053.0
2           0.0             3281.0            1355.0      11870.0
3           0.0              111.0              30.0        523.0
4           0.0             5011.0            1515.0      18682.0

   stats.stddev  stats.subtype.boardgame.bayesaverage  \
0       1.59321                               7.29168
1       1.46282                               5.87150
2       1.18531                               7.28295
3       1.21028                               5.76636
4       1.33020                               7.21895

   stats.subtype.boardgame.pos  stats.trading  stats.usersrated  \
0                        147.0          170.0            4498.0
1                       2541.0           73.0             478.0
2                        150.0          234.0           12019.0
3                       3191.0           29.0             314.0
4                        181.0          823.0           15195.0

   stats.wanting  stats.wishing polls.language_dependence  \
0          505.0         1654.0                        No
1           67.0          161.0                      Some
```

```
2          707.0         2601.0                         No
3           61.0          112.0                         No
4          516.0         2219.0                         No

  polls.suggested_numplayers.1 polls.suggested_numplayers.10  \
0               NotRecommended                           None
1               NotRecommended                           None
2               NotRecommended                           None
3               NotRecommended                           None
4               NotRecommended                           None

  polls.suggested_numplayers.2 polls.suggested_numplayers.3  \
0               NotRecommended                NotRecommended
1               NotRecommended                   Recommended
2                  Recommended                          Best
3                  Recommended                          Best
4               NotRecommended                   Recommended

  polls.suggested_numplayers.4 polls.suggested_numplayers.5  \
0                  Recommended                          Best
1                         Best                          None
2                  Recommended                          None
3                         Best                          None
4                         Best                   Recommended

  polls.suggested_numplayers.6 polls.suggested_numplayers.7  \
0                         None                          None
1                         None                          None
2                         None                          None
3                         None                          None
4                  Recommended                          None

  polls.suggested_numplayers.8 polls.suggested_numplayers.9  \
0                         None                          None
1                         None                          None
2                         None                          None
3                         None                          None
4                         None                          None

  polls.suggested_numplayers.Over polls.suggested_playerage  \
0                  NotRecommended                         14
1                  NotRecommended                       None
2                  NotRecommended                         10
3                  NotRecommended                         14
4                  NotRecommended                         12

  attributes.t.links.concat.2...  stats.family.amiga.bayesaverage  \
0                           None                              NaN
```

```
1                            None                                  NaN
2                            None                                  NaN
3                            None                                  NaN
4                            None                                  NaN

   stats.family.amiga.pos  stats.family.arcade.bayesaverage  \
0                     NaN                               NaN
1                     NaN                               NaN
2                     NaN                               NaN
3                     NaN                               NaN
4                     NaN                               NaN

   stats.family.arcade.pos  stats.family.atarist.bayesaverage  \
0                      NaN                                NaN
1                      NaN                                NaN
2                      NaN                                NaN
3                      NaN                                NaN
4                      NaN                                NaN

   stats.family.atarist.pos  stats.family.commodore64.bayesaverage  \
0                       NaN                                    NaN
1                       NaN                                    NaN
2                       NaN                                    NaN
3                       NaN                                    NaN
4                       NaN                                    NaN

   stats.family.commodore64.pos  stats.subtype.rpgitem.bayesaverage  \
0                           NaN                                 NaN
1                           NaN                                 NaN
2                           NaN                                 NaN
3                           NaN                                 NaN
4                           NaN                                 NaN

   stats.subtype.rpgitem.pos  stats.subtype.videogame.bayesaverage  \
0                        NaN                                   NaN
1                        NaN                                   NaN
2                        NaN                                   NaN
3                        NaN                                   NaN
4                        NaN                                   NaN

   stats.subtype.videogame.pos
0                          NaN
1                          NaN
2                          NaN
3                          NaN
4                          NaN
```

In [8]: *# Describe the data frame fields and contents*

```
df_bg.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 90400 entries, 0 to 90399
Data columns (total 81 columns):
row_names                                    90400 non-null object
game.id                                      90400 non-null object
game.type                                    90400 non-null object
details.description                          90400 non-null object
details.image                                83508 non-null object
details.maxplayers                           90397 non-null float64
details.maxplaytime                          90397 non-null float64
details.minage                               90397 non-null float64
details.minplayers                           90397 non-null float64
details.minplaytime                          90397 non-null float64
details.name                                 90400 non-null object
details.playingtime                          90397 non-null float64
details.thumbnail                            83508 non-null object
details.yearpublished                        90397 non-null float64
attributes.boardgameartist                   34665 non-null object
attributes.boardgamecategory                 88869 non-null object
attributes.boardgamecompilation              2295 non-null object
attributes.boardgamedesigner                 78517 non-null object
attributes.boardgameexpansion                17787 non-null object
attributes.boardgamefamily                   50168 non-null object
attributes.boardgameimplementation           4688 non-null object
attributes.boardgameintegration              1284 non-null object
attributes.boardgamemechanic                 75163 non-null object
attributes.boardgamepublisher                90305 non-null object
attributes.total                             90400 non-null float64
stats.average                                90400 non-null float64
stats.averageweight                          90400 non-null float64
stats.bayesaverage                           90400 non-null float64
stats.family.abstracts.bayesaverage          897 non-null float64
stats.family.abstracts.pos                   858 non-null float64
stats.family.cgs.bayesaverage                342 non-null float64
stats.family.cgs.pos                         264 non-null float64
stats.family.childrensgames.bayesaverage     679 non-null float64
stats.family.childrensgames.pos              670 non-null float64
stats.family.familygames.bayesaverage        1712 non-null float64
stats.family.familygames.pos                 1558 non-null float64
stats.family.partygames.bayesaverage         476 non-null float64
stats.family.partygames.pos                  436 non-null float64
stats.family.strategygames.bayesaverage      2002 non-null float64
stats.family.strategygames.pos               1631 non-null float64
stats.family.thematic.bayesaverage           1127 non-null float64
stats.family.thematic.pos                    870 non-null float64
stats.family.wargames.bayesaverage           3036 non-null float64
```

```
stats.family.wargames.pos                    2530 non-null float64
stats.median                                 90400 non-null float64
stats.numcomments                            90400 non-null float64
stats.numweights                             90400 non-null float64
stats.owned                                  90400 non-null float64
stats.stddev                                 90400 non-null float64
stats.subtype.boardgame.bayesaverage         18064 non-null float64
stats.subtype.boardgame.pos                  13693 non-null float64
stats.trading                                90400 non-null float64
stats.usersrated                             90400 non-null float64
stats.wanting                                90400 non-null float64
stats.wishing                                90400 non-null float64
polls.language_dependence                    20728 non-null object
polls.suggested_numplayers.1                 17478 non-null object
polls.suggested_numplayers.10                866 non-null object
polls.suggested_numplayers.2                 20381 non-null object
polls.suggested_numplayers.3                 14809 non-null object
polls.suggested_numplayers.4                 15102 non-null object
polls.suggested_numplayers.5                 8522 non-null object
polls.suggested_numplayers.6                 5984 non-null object
polls.suggested_numplayers.7                 2264 non-null object
polls.suggested_numplayers.8                 1982 non-null object
polls.suggested_numplayers.9                 920 non-null object
polls.suggested_numplayers.Over              16175 non-null object
polls.suggested_playerage                    18205 non-null object
attributes.t.links.concat.2...              65 non-null object
stats.family.amiga.bayesaverage              1 non-null float64
stats.family.amiga.pos                       1 non-null float64
stats.family.arcade.bayesaverage             1 non-null float64
stats.family.arcade.pos                      1 non-null float64
stats.family.atarist.bayesaverage            1 non-null float64
stats.family.atarist.pos                     1 non-null float64
stats.family.commodore64.bayesaverage        1 non-null float64
stats.family.commodore64.pos                 1 non-null float64
stats.subtype.rpgitem.bayesaverage           4 non-null float64
stats.subtype.rpgitem.pos                    2 non-null float64
stats.subtype.videogame.bayesaverage         1 non-null float64
stats.subtype.videogame.pos                  1 non-null float64
dtypes: float64(50), object(31)
memory usage: 55.9+ MB
```

We want to filter out a few types of entries:
(1) Those with fewer than 10 rating entries [where the results may be biased by a few users]
(2) Board Game expansions [we will only consider base games here]
(3) Video games & role-playing books [this analysis focuses on table-top board games]

We can accomplish this by filtering on the following variables:
- game.type (filter out non-Board Games to remove expansions) - stats.usersrated (Filter out less

than 10) - stats.subtype.rpgitem.pos (Filter out) - stats.subtype.videogame.pos (Filter out)

```
In [9]: # Filter out games with fewer than 10 rating entries
        # (also filter for published year > 1900 to remove irrelevant data)
        df_filter = df_bg[(df_bg['stats.usersrated'] > 10) &
                          (df_bg['details.yearpublished'] > 1900)]

        # Filter out expansions
        df_filter = df_filter[(df_filter['game.type']) == "boardgame"]
        # NOTE: expansions have game.type "boardgameexpansion"

        # Filter out video games and role-playing books
        df_filter = df_filter[(df_filter['stats.subtype.rpgitem.pos'].isnull()) &
                              (df_filter['stats.subtype.videogame.pos'].isnull())]

        # Final filtered data frame:
        df_filter.describe()
```

Out[9]:

	details.maxplayers	details.maxplaytime	details.minage
count	20842.000000	20842.000000	20842.000000
mean	5.341618	76.394828	8.960608
std	14.373976	488.914626	3.997340
min	0.000000	0.000000	0.000000
25%	2.000000	20.000000	8.000000
50%	4.000000	40.000000	10.000000
75%	6.000000	90.000000	12.000000
max	999.000000	60000.000000	42.000000

	details.minplayers	details.minplaytime	details.playingtime
count	20842.000000	20842.000000	20842.000000
mean	2.054937	67.869590	76.394828
std	0.688318	461.959652	488.914626
min	0.000000	0.000000	0.000000
25%	2.000000	20.000000	20.000000
50%	2.000000	30.000000	40.000000
75%	2.000000	60.000000	90.000000
max	10.000000	60000.000000	60000.000000

	details.yearpublished	attributes.total	stats.average
count	20842.000000	20842.000000	20842.000000
mean	2002.278860	5.483351	6.177499
std	13.800454	1.205738	1.008343
min	1901.000000	1.000000	1.226530
25%	1996.000000	5.000000	5.556000
50%	2007.000000	5.000000	6.219970
75%	2012.000000	6.000000	6.861903
max	2018.000000	10.000000	9.411770

```
       stats.averageweight  stats.bayesaverage  \
count         20842.000000        20842.000000
mean              1.869084            3.747665
std               0.924572            2.737987
min               0.000000            0.000000
25%               1.181800            0.000000
50%               1.833300            5.510885
75%               2.500000            5.617965
max               5.000000            8.489660

       stats.family.abstracts.bayesaverage  stats.family.abstracts.pos  \
count                           806.000000                  798.000000
mean                              5.829950                  425.190476
std                               0.439721                  245.290075
min                               4.698510                    1.000000
25%                               5.538267                  212.500000
50%                               5.729110                  422.500000
75%                               6.046375                  637.750000
max                               7.649620                  855.000000

       stats.family.cgs.bayesaverage  stats.family.cgs.pos  \
count                     266.000000            264.000000
mean                        5.963841            132.500000
std                         0.749003             76.354437
min                         3.627810              1.000000
25%                         5.463457             66.750000
50%                         5.823545            132.500000
75%                         6.366967            198.250000
max                         8.040870            264.000000

       stats.family.childrensgames.bayesaverage  \
count                                650.000000
mean                                   5.490099
std                                    0.560599
min                                    2.871180
25%                                    5.188495
50%                                    5.490630
75%                                    5.832482
max                                    6.951310

       stats.family.childrensgames.pos  stats.family.familygames.bayesaverage  \
count                       650.000000                            1547.000000
mean                        327.836923                               5.986206
std                         190.252685                               0.484513
min                           1.000000                               4.093260
25%                         163.250000                               5.631495
50%                         326.500000                               5.892270
75%                         491.750000                               6.266205
```

```
max                          667.000000                                7.863330

       stats.family.familygames.pos  stats.family.partygames.bayesaverage  \
count                   1532.000000                            431.000000
mean                     777.039817                              5.955147
std                      447.814427                              0.580266
min                        1.000000                              3.383350
25%                      389.750000                              5.586265
50%                      778.500000                              5.886040
75%                     1163.250000                              6.249760
max                     1556.000000                              7.827470

       stats.family.partygames.pos  stats.family.strategygames.bayesaverage  \
count                   426.000000                             1645.000000
mean                    215.969484                                6.299888
std                     124.543124                                0.602057
min                       1.000000                                4.915410
25%                     109.250000                                5.795290
50%                     215.500000                                6.184040
75%                     322.750000                                6.694980
max                     435.000000                                8.479230

       stats.family.strategygames.pos  stats.family.thematic.bayesaverage  \
count                     1625.000000                          879.000000
mean                       815.225846                            6.156519
std                        471.140369                            0.610644
min                          1.000000                            4.850390
25%                        407.000000                            5.677385
50%                        815.000000                            5.989790
75%                       1223.000000                            6.521710
max                       1630.000000                            8.494430

       stats.family.thematic.pos  stats.family.wargames.bayesaverage  \
count                 870.000000                          2555.000000
mean                  435.301149                             6.017413
std                   251.027497                             0.494920
min                     1.000000                             4.494430
25%                   218.250000                             5.670540
50%                   435.500000                             5.900000
75%                   652.750000                             6.248730
max                   869.000000                             8.305610

       stats.family.wargames.pos  stats.median  stats.numcomments  \
count                2523.000000       20842.0       20842.000000
mean                 1263.164487           0.0         131.236829
std                   730.036972           0.0         464.393914
min                     1.000000           0.0           0.000000
25%                   631.500000           0.0          10.000000
```

```
50%                 1262.000000           0.0          25.000000
75%                 1893.500000           0.0          74.000000
max                 2530.000000           0.0       13841.000000

       stats.numweights   stats.owned  stats.stddev  \
count      20842.000000  20842.000000  20842.000000
mean          38.892429    729.420257      1.518226
std          173.669855   2842.386204      0.343919
min            0.000000      0.000000      0.369966
25%            2.000000     54.000000      1.296360
50%            6.000000    136.000000      1.470660
75%           18.000000    417.750000      1.684810
max         6448.000000  95401.000000      3.968040

       stats.subtype.boardgame.bayesaverage  stats.subtype.boardgame.pos  \
count                          13656.000000                 13483.000000
mean                               5.720150                  6814.585552
std                                0.398324                  3940.078844
min                                3.713330                     1.000000
25%                                5.513438                  3401.000000
50%                                5.569265                  6811.000000
75%                                5.743003                 10220.500000
max                                8.489660                 13680.000000

       stats.trading  stats.usersrated  stats.wanting  stats.wishing  \
count   20842.000000      20842.000000   20842.000000   20842.000000
mean       27.322954        457.413588      30.181077     117.509116
std        70.036042       2102.773633      98.820719     447.580727
min         0.000000         11.000000       0.000000       0.000000
25%         2.000000         21.000000       1.000000       5.000000
50%         7.000000         52.000000       5.000000      14.000000
75%        22.000000        182.000000      17.000000      53.000000
max      1858.000000      67655.000000    1838.000000    9082.000000

       stats.family.amiga.bayesaverage  stats.family.amiga.pos  \
count                              0.0                     0.0
mean                               NaN                     NaN
std                                NaN                     NaN
min                                NaN                     NaN
25%                                NaN                     NaN
50%                                NaN                     NaN
75%                                NaN                     NaN
max                                NaN                     NaN

       stats.family.arcade.bayesaverage  stats.family.arcade.pos  \
count                               0.0                      0.0
mean                                NaN                      NaN
std                                 NaN                      NaN
```

```
min                                    NaN                       NaN
25%                                    NaN                       NaN
50%                                    NaN                       NaN
75%                                    NaN                       NaN
max                                    NaN                       NaN

       stats.family.atarist.bayesaverage  stats.family.atarist.pos  \
count                                0.0                       0.0
mean                                 NaN                       NaN
std                                  NaN                       NaN
min                                  NaN                       NaN
25%                                  NaN                       NaN
50%                                  NaN                       NaN
75%                                  NaN                       NaN
max                                  NaN                       NaN

       stats.family.commodore64.bayesaverage  stats.family.commodore64.pos  \
count                                    0.0                           0.0
mean                                     NaN                           NaN
std                                      NaN                           NaN
min                                      NaN                           NaN
25%                                      NaN                           NaN
50%                                      NaN                           NaN
75%                                      NaN                           NaN
max                                      NaN                           NaN

       stats.subtype.rpgitem.bayesaverage  stats.subtype.rpgitem.pos  \
count                                 0.0                        0.0
mean                                  NaN                        NaN
std                                   NaN                        NaN
min                                   NaN                        NaN
25%                                   NaN                        NaN
50%                                   NaN                        NaN
75%                                   NaN                        NaN
max                                   NaN                        NaN

       stats.subtype.videogame.bayesaverage  stats.subtype.videogame.pos
count                                   0.0                          0.0
mean                                    NaN                          NaN
std                                     NaN                          NaN
min                                     NaN                          NaN
25%                                     NaN                          NaN
50%                                     NaN                          NaN
75%                                     NaN                          NaN
max                                     NaN                          NaN
```

1.3.2 Processing Feature Variables

As discussed above, this analysis focuses on the aspects of a board game that designers can directly control. Those features are given in the variables below:

Features

- details.maxplayers
- details.minage
- details.minplayers
- details.playingtime
- details.yearpublished
- attributes.boardgamecategory (will require pre-processing)
- attributes.boardgamemechanic (will require pre-processing)
- stats.averageweight (NOTE: this is a sort of complexity rating for gameplay)

Other Fields

- We'll keep details.name to be able to refer to specific games

A bit of pre-processing is required to make the Category and Mechanic fields more usable. The given database records these things in comma-separated lists, whereas we need a set of binary fields (one per category and mechanic). The cells below perform the preprocessing on the boardgamecategory and boardgamemechanic variables.

```
In [10]: # Next we'll try to pre-process attributes.boardgamemechanic

         df_mechs = df_filter['attributes.boardgamemechanic'].str.get_dummies(sep=',')

In [11]: df_mechs.head()
```

Out[11]:

	Acting	Action / Movement Programming	Action Point Allowance System	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	1	
4	0	0	0	

	Area Control / Area Influence	Area Enclosure	Area Movement	Area-Impulse	\
0	1	0	0	0	
1	0	0	0	0	
2	1	0	0	0	
3	1	0	0	0	
4	0	0	0	0	

	Auction/Bidding	Betting/Wagering	Campaign / Battle Card Driven	\
0	1	0	0	
1	0	0	0	
2	0	0	0	
3	1	0	0	

```
4                0                0                              0

   Card Drafting  Chit-Pull System  Co-operative Play  Commodity Speculation  \
0              0                 0                  0                      0
1              0                 0                  0                      0
2              0                 0                  0                      0
3              0                 0                  0                      0
4              0                 0                  0                      0

   Crayon Rail System  Deck / Pool Building  Dice Rolling  Grid Movement  \
0                   0                     0             1              0
1                   0                     0             0              0
2                   0                     0             0              0
3                   0                     0             0              0
4                   0                     0             0              0

   Hand Management  Hex-and-Counter  Line Drawing  Memory  Modular Board  \
0                1                0             0       0              0
1                0                0             0       0              0
2                1                0             0       0              0
3                0                0             0       0              0
4                1                0             0       0              0

   Paper-and-Pencil  Partnerships  Pattern Building  Pattern Recognition  \
0                 0             0                 0                    0
1                 0             0                 0                    0
2                 0             0                 0                    0
3                 0             0                 0                    0
4                 0             0                 0                    0

   Pick-up and Deliver  Player Elimination  Point to Point Movement  \
0                    0                   0                        0
1                    0                   0                        0
2                    0                   0                        0
3                    0                   0                        0
4                    0                   0                        0

   Press Your Luck  Rock-Paper-Scissors  Role Playing  Roll / Spin and Move  \
0                0                    0             0                     0
1                0                    0             0                     0
2                0                    0             0                     0
3                0                    0             0                     0
4                0                    0             0                     0

   Route/Network Building  Secret Unit Deployment  Set Collection  Simulation  \
0                       0                       0               0           0
1                       0                       0               0           0
2                       0                       0               1           0
```

```
3                       0                     0              1           0
4                       0                     0              0           0

   Simultaneous Action Selection  Singing  Stock Holding  Storytelling  \
0                              1        0              0             0
1                              0        0              0             0
2                              0        0              0             0
3                              0        0              0             0
4                              0        0              1             0

   Take That  Tile Placement  Time Track  Trading  Trick-taking  \
0          0               0           0        0             0
1          0               0           0        0             1
2          0               1           0        0             0
3          0               0           0        0             0
4          0               1           0        0             0

   Variable Phase Order  Variable Player Powers  Voting  Worker Placement
0                     0                       0       0                 0
1                     0                       0       0                 0
2                     0                       0       0                 0
3                     0                       0       0                 0
4                     0                       0       0                 0
```

In [12]: df_mechs.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 20842 entries, 0 to 90327
Data columns (total 51 columns):
Acting                          20842 non-null int64
Action / Movement Programming   20842 non-null int64
Action Point Allowance System   20842 non-null int64
Area Control / Area Influence   20842 non-null int64
Area Enclosure                  20842 non-null int64
Area Movement                   20842 non-null int64
Area-Impulse                    20842 non-null int64
Auction/Bidding                 20842 non-null int64
Betting/Wagering                20842 non-null int64
Campaign / Battle Card Driven   20842 non-null int64
Card Drafting                   20842 non-null int64
Chit-Pull System                20842 non-null int64
Co-operative Play               20842 non-null int64
Commodity Speculation           20842 non-null int64
Crayon Rail System              20842 non-null int64
Deck / Pool Building            20842 non-null int64
Dice Rolling                    20842 non-null int64
Grid Movement                   20842 non-null int64
Hand Management                 20842 non-null int64
```

```
Hex-and-Counter                  20842 non-null int64
Line Drawing                     20842 non-null int64
Memory                           20842 non-null int64
Modular Board                    20842 non-null int64
Paper-and-Pencil                 20842 non-null int64
Partnerships                     20842 non-null int64
Pattern Building                 20842 non-null int64
Pattern Recognition              20842 non-null int64
Pick-up and Deliver              20842 non-null int64
Player Elimination               20842 non-null int64
Point to Point Movement          20842 non-null int64
Press Your Luck                  20842 non-null int64
Rock-Paper-Scissors              20842 non-null int64
Role Playing                     20842 non-null int64
Roll / Spin and Move             20842 non-null int64
Route/Network Building           20842 non-null int64
Secret Unit Deployment           20842 non-null int64
Set Collection                   20842 non-null int64
Simulation                       20842 non-null int64
Simultaneous Action Selection    20842 non-null int64
Singing                          20842 non-null int64
Stock Holding                    20842 non-null int64
Storytelling                     20842 non-null int64
Take That                        20842 non-null int64
Tile Placement                   20842 non-null int64
Time Track                       20842 non-null int64
Trading                          20842 non-null int64
Trick-taking                     20842 non-null int64
Variable Phase Order             20842 non-null int64
Variable Player Powers           20842 non-null int64
Voting                           20842 non-null int64
Worker Placement                 20842 non-null int64
dtypes: int64(51)
memory usage: 8.3 MB
```

```
In [13]: mechs_selector = list(df_mechs)
```

```
In [14]: df_cats = df_filter['attributes.boardgamecategory'].str.get_dummies(sep=',')
```

```
In [15]: df_cats.head()
```

```
Out[15]:    Abstract Strategy  Action / Dexterity  Adventure  Age of Reason  \
         0                  0                   0          0              0
         1                  0                   0          0              0
         2                  1                   0          0              0
         3                  0                   0          0              0
         4                  0                   0          0              0
```

	American Civil War	American Indian Wars	American Revolutionary War
0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0

	American West	Ancient	Animals	Arabian	Aviation / Flight	Bluffing
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	1	0	0	0	0
4	0	0	0	0	0	0

	Book	Card Game	Children's Game	City Building	Civil War	Civilization
0	0	0	0	0	0	0
1	0	1	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

	Collectible Components	Comic Book / Strip	Deduction	Dice	Economic
0	0	0	0	0	1
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	1

	Educational	Electronic	Environmental	Expansion for Base-game
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0

	Exploration	Fan Expansion	Fantasy	Farming	Fighting	Game System
0	0	0	0	0	0	0
1	0	0	1	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

	Horror	Humor	Industry / Manufacturing	Korean War	Mafia	Math
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

```
   Mature / Adult  Maze  Medical  Medieval  Memory  Miniatures  \
0               0     0        0         0       0           0
1               0     0        0         0       0           0
2               0     0        0         1       0           0
3               0     0        0         0       0           0
4               0     0        0         0       0           0

   Modern Warfare  Movies / TV / Radio theme  Murder/Mystery  Music  \
0               0                          0               0      0
1               0                          0               0      0
2               0                          0               0      0
3               0                          0               0      0
4               0                          0               0      0

   Mythology  Napoleonic  Nautical  Negotiation  Novel-based  Number  \
0          0           0         0            1            0       0
1          0           0         0            0            0       0
2          0           0         0            0            0       0
3          0           0         0            0            0       0
4          0           0         0            0            0       0

   Party Game  Pike and Shot  Pirates  Political  Post-Napoleonic  \
0           0              0        0          1                0
1           0              0        0          0                0
2           0              0        0          0                0
3           0              0        0          0                0
4           0              0        0          0                0

   Prehistoric  Print & Play  Puzzle  Racing  Real-time  Religious  \
0            0             0       0       0          0          0
1            0             0       0       0          0          0
2            0             0       0       0          0          0
3            0             0       0       0          0          0
4            0             0       0       0          0          0

   Renaissance  Science Fiction  Space Exploration  Spies/Secret Agents  \
0            0                0                  0                    0
1            0                0                  0                    0
2            0                0                  0                    0
3            0                0                  0                    0
4            0                0                  0                    0

   Sports  Territory Building  Trains  Transportation  Travel  Trivia  \
0       0                   0       0               0       0       0
1       0                   0       0               0       0       0
2       0                   0       0               0       0       0
3       0                   0       0               0       0       0
```

```
4       0                   0       0               0       0       0

   Video Game Theme  Vietnam War  Wargame  Word Game  World War I  \
0                 0            0        0          0            0
1                 0            0        0          0            0
2                 0            0        0          0            0
3                 0            0        0          0            0
4                 0            0        0          0            0

   World War II  Zombies
0             0        0
1             0        0
2             0        0
3             0        0
4             0        0
```

In [16]: `df_cats.info()`

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 20842 entries, 0 to 90327
Data columns (total 84 columns):
Abstract Strategy             20842 non-null int64
Action / Dexterity            20842 non-null int64
Adventure                     20842 non-null int64
Age of Reason                 20842 non-null int64
American Civil War            20842 non-null int64
American Indian Wars          20842 non-null int64
American Revolutionary War    20842 non-null int64
American West                 20842 non-null int64
Ancient                       20842 non-null int64
Animals                       20842 non-null int64
Arabian                       20842 non-null int64
Aviation / Flight             20842 non-null int64
Bluffing                      20842 non-null int64
Book                          20842 non-null int64
Card Game                     20842 non-null int64
Children's Game               20842 non-null int64
City Building                 20842 non-null int64
Civil War                     20842 non-null int64
Civilization                  20842 non-null int64
Collectible Components        20842 non-null int64
Comic Book / Strip            20842 non-null int64
Deduction                     20842 non-null int64
Dice                          20842 non-null int64
Economic                      20842 non-null int64
Educational                   20842 non-null int64
Electronic                    20842 non-null int64
Environmental                 20842 non-null int64
```

```
Expansion for Base-game       20842 non-null int64
Exploration                   20842 non-null int64
Fan Expansion                 20842 non-null int64
Fantasy                       20842 non-null int64
Farming                       20842 non-null int64
Fighting                      20842 non-null int64
Game System                   20842 non-null int64
Horror                        20842 non-null int64
Humor                         20842 non-null int64
Industry / Manufacturing      20842 non-null int64
Korean War                    20842 non-null int64
Mafia                         20842 non-null int64
Math                          20842 non-null int64
Mature / Adult                20842 non-null int64
Maze                          20842 non-null int64
Medical                       20842 non-null int64
Medieval                      20842 non-null int64
Memory                        20842 non-null int64
Miniatures                    20842 non-null int64
Modern Warfare                20842 non-null int64
Movies / TV / Radio theme     20842 non-null int64
Murder/Mystery                20842 non-null int64
Music                         20842 non-null int64
Mythology                     20842 non-null int64
Napoleonic                    20842 non-null int64
Nautical                      20842 non-null int64
Negotiation                   20842 non-null int64
Novel-based                   20842 non-null int64
Number                        20842 non-null int64
Party Game                    20842 non-null int64
Pike and Shot                 20842 non-null int64
Pirates                       20842 non-null int64
Political                     20842 non-null int64
Post-Napoleonic               20842 non-null int64
Prehistoric                   20842 non-null int64
Print & Play                  20842 non-null int64
Puzzle                        20842 non-null int64
Racing                        20842 non-null int64
Real-time                     20842 non-null int64
Religious                     20842 non-null int64
Renaissance                   20842 non-null int64
Science Fiction               20842 non-null int64
Space Exploration             20842 non-null int64
Spies/Secret Agents           20842 non-null int64
Sports                        20842 non-null int64
Territory Building            20842 non-null int64
Trains                        20842 non-null int64
Transportation                20842 non-null int64
```

```
Travel                        20842 non-null int64
Trivia                        20842 non-null int64
Video Game Theme              20842 non-null int64
Vietnam War                   20842 non-null int64
Wargame                       20842 non-null int64
Word Game                     20842 non-null int64
World War I                   20842 non-null int64
World War II                  20842 non-null int64
Zombies                       20842 non-null int64
dtypes: int64(84)
memory usage: 13.5 MB
```

```
In [17]: cats_selector = list(df_cats)
```

```
In [18]: #Now let's create a new Dataframe with only the features we want to keep
         df_limit = df_filter[['details.name','details.maxplayers', 'details.minage', 'details
```

```
In [19]: df_limit.head(10)
```

Out[19]:

	details.name	details.maxplayers	details.minage	details.minplayers	\
0	Die Macher	5.0	14.0	3.0	
1	Dragonmaster	4.0	12.0	3.0	
2	Samurai	4.0	10.0	2.0	
3	Tal der Könige	4.0	12.0	2.0	
4	Acquire	6.0	12.0	3.0	
5	Mare Mediterraneum	6.0	12.0	2.0	
6	Cathedral	2.0	8.0	2.0	
7	Lords of Creation	5.0	12.0	2.0	
8	El Caballero	4.0	13.0	2.0	
9	Elfenland	6.0	10.0	2.0	

	details.playingtime	details.yearpublished	stats.averageweight	\
0	240.0	1986.0	4.3477	
1	30.0	1981.0	1.9423	
2	60.0	1998.0	2.5085	
3	60.0	1992.0	2.6667	
4	90.0	1964.0	2.5089	
5	240.0	1989.0	3.0000	
6	20.0	1978.0	1.8217	
7	120.0	1993.0	2.4000	
8	90.0	1998.0	3.1958	
9	60.0	1998.0	2.1649	

	stats.average	stats.owned	stats.stddev
0	7.66508	5251.0	1.59321
1	6.60815	1053.0	1.46282
2	7.44119	11870.0	1.18531
3	6.60675	523.0	1.21028

```
4        7.35830      18682.0       1.33020
5        6.52534        106.0       1.65064
6        6.50534       4561.0       1.31078
7        6.14538        460.0       1.25508
8        6.51776       2263.0       1.40413
9        6.74996       7792.0       1.24616
```

```
In [20]: df_interim = df_cats.join(df_mechs, lsuffix='_category', rsuffix='_mechanics')
```

```
In [21]: df_interim.head()
```

Out[21]:

	Abstract Strategy	Action / Dexterity	Adventure	Age of Reason	\
0	0	0	0	0	
1	0	0	0	0	
2	1	0	0	0	
3	0	0	0	0	
4	0	0	0	0	

	American Civil War	American Indian Wars	American Revolutionary War	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	

	American West	Ancient	Animals	Arabian	Aviation / Flight	Bluffing	\
0	0	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	1	0	0	0	0	
4	0	0	0	0	0	0	

	Book	Card Game	Children's Game	City Building	Civil War	Civilization	\
0	0	0	0	0	0	0	
1	0	1	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	

	Collectible Components	Comic Book / Strip	Deduction	Dice	Economic	\
0	0	0	0	0	1	
1	0	0	0	0	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	1	

	Educational	Electronic	Environmental	Expansion for Base-game	\
0	0	0	0	0	

```
1            0           0              0                      0
2            0           0              0                      0
3            0           0              0                      0
4            0           0              0                      0

   Exploration  Fan Expansion  Fantasy  Farming  Fighting  Game System  \
0            0              0        0        0         0            0
1            0              0        1        0         0            0
2            0              0        0        0         0            0
3            0              0        0        0         0            0
4            0              0        0        0         0            0

   Horror  Humor  Industry / Manufacturing  Korean War  Mafia  Math  \
0       0      0                         0           0      0     0
1       0      0                         0           0      0     0
2       0      0                         0           0      0     0
3       0      0                         0           0      0     0
4       0      0                         0           0      0     0

   Mature / Adult  Maze  Medical  Medieval  Memory_category  Miniatures  \
0               0     0        0         0                0           0
1               0     0        0         0                0           0
2               0     0        0         1                0           0
3               0     0        0         0                0           0
4               0     0        0         0                0           0

   Modern Warfare  Movies / TV / Radio theme  Murder/Mystery  Music  \
0               0                          0               0      0
1               0                          0               0      0
2               0                          0               0      0
3               0                          0               0      0
4               0                          0               0      0

         ...         Action / Movement Programming  \
0        ...                                     0
1        ...                                     0
2        ...                                     0
3        ...                                     0
4        ...                                     0

   Action Point Allowance System  Area Control / Area Influence  \
0                              0                               1
1                              0                               0
2                              0                               1
3                              1                               1
4                              0                               0

   Area Enclosure  Area Movement  Area-Impulse  Auction/Bidding  \
```

```
0               0             0             0               1
1               0             0             0               0
2               0             0             0               0
3               0             0             0               1
4               0             0             0               0

   Betting/Wagering  Campaign / Battle Card Driven  Card Drafting  \
0                 0                              0              0
1                 0                              0              0
2                 0                              0              0
3                 0                              0              0
4                 0                              0              0

   Chit-Pull System  Co-operative Play  Commodity Speculation  \
0                 0                  0                      0
1                 0                  0                      0
2                 0                  0                      0
3                 0                  0                      0
4                 0                  0                      0

   Crayon Rail System  Deck / Pool Building  Dice Rolling  Grid Movement  \
0                   0                     0             1              0
1                   0                     0             0              0
2                   0                     0             0              0
3                   0                     0             0              0
4                   0                     0             0              0

   Hand Management  Hex-and-Counter  Line Drawing  Memory_mechanics  \
0                1                0             0                 0
1                0                0             0                 0
2                1                0             0                 0
3                0                0             0                 0
4                1                0             0                 0

   Modular Board  Paper-and-Pencil  Partnerships  Pattern Building  \
0              0                 0             0                 0
1              0                 0             0                 0
2              0                 0             0                 0
3              0                 0             0                 0
4              0                 0             0                 0

   Pattern Recognition  Pick-up and Deliver  Player Elimination  \
0                    0                    0                   0
1                    0                    0                   0
2                    0                    0                   0
3                    0                    0                   0
4                    0                    0                   0
```

```
   Point to Point Movement  Press Your Luck  Rock-Paper-Scissors  \
0                        0                0                    0
1                        0                0                    0
2                        0                0                    0
3                        0                0                    0
4                        0                0                    0

   Role Playing  Roll / Spin and Move  Route/Network Building  \
0             0                     0                       0
1             0                     0                       0
2             0                     0                       0
3             0                     0                       0
4             0                     0                       0

   Secret Unit Deployment  Set Collection  Simulation  \
0                       0               0           0
1                       0               0           0
2                       0               1           0
3                       0               1           0
4                       0               0           0

   Simultaneous Action Selection  Singing  Stock Holding  Storytelling  \
0                              1        0              0             0
1                              0        0              0             0
2                              0        0              0             0
3                              0        0              0             0
4                              0        0              1             0

   Take That  Tile Placement  Time Track  Trading  Trick-taking  \
0          0               0           0        0             0
1          0               0           0        0             1
2          0               1           0        0             0
3          0               0           0        0             0
4          0               1           0        0             0

   Variable Phase Order  Variable Player Powers  Voting  Worker Placement
0                     0                       0       0                 0
1                     0                       0       0                 0
2                     0                       0       0                 0
3                     0                       0       0                 0
4                     0                       0       0                 0

[5 rows x 135 columns]
```

In [22]:
```
df_data = df_limit.join(df_interim)
```

In [23]:
```
df_data
```

Out[23]:
```
                                        details.name  details.maxplayers  \
```

0	Die Macher	5.0
1	Dragonmaster	4.0
2	Samurai	4.0
3	Tal der Könige	4.0
4	Acquire	6.0
5	Mare Mediterraneum	6.0
6	Cathedral	2.0
7	Lords of Creation	5.0
8	El Caballero	4.0
9	Elfenland	6.0
10	Bohnanza	7.0
11	Ra	5.0
12	Catan	4.0
13	Basari	4.0
14	Cosmic Encounter	6.0
15	MarraCash	4.0
16	Button Men	2.0
17	RoboRally	8.0
18	Wacky Wacky West	4.0
19	Full Metal Planète	4.0
20	Gateway to the Stars	7.0
21	Magic Realm	16.0
22	Divine Right	6.0
23	Twilight Imperium	6.0
24	Battlemist	6.0
25	Age of Renaissance	6.0
26	Supremacy	6.0
27	Illuminati: Deluxe Edition	8.0
28	Terrain Vague	4.0
29	Dark Tower	4.0
...	...	...
89526	Tenno	7.0
89536	Tembo	4.0
89537	NMBR 9	4.0
89546	Qwinto: Das Kartenspiel	4.0
89547	Teenage Mutant Ninja Turtles Dice Masters: Her...	4.0
89549	Twenty One	6.0
89570	Vanguard of War	4.0
89574	Go Go Gelato!	4.0
89621	Carry On: Gear Collecting Card Game	4.0
89630	Import / Export	6.0
89668	Delve	4.0
89697	Napoleons Quagmire	2.0
89727	Profiler	8.0
89778	Rise to Nobility	6.0
89786	Deckscape: Test Time	6.0
89788	3 Secrets	8.0
89821	Aeon's End: War Eternal	4.0

```
89852                          Empires of the Void II                5.0
89856                                   Tortuga 1667                 9.0
89881                                 Divinity Derby                 6.0
89962                Game of Quotes  Verrückte Zitate                6.0
89969                                     Yukon Salon                 4.0
90001                                       Valletta                  4.0
90006           Warhammer Quest: Shadows Over Hammerhal               5.0
90031                                   Santo Domingo                 6.0
90065                                 National Economy                4.0
90167                                          Solaris                5.0
90206                                        Math Fluxx               6.0
90239                                        Bärenpark                4.0
90327                    Vikings Gone Wild Ultimate Set               5.0

       details.minage  details.minplayers  details.playingtime  \
0                14.0                 3.0                240.0
1                12.0                 3.0                 30.0
2                10.0                 2.0                 60.0
3                12.0                 2.0                 60.0
4                12.0                 3.0                 90.0
5                12.0                 2.0                240.0
6                 8.0                 2.0                 20.0
7                12.0                 2.0                120.0
8                13.0                 2.0                 90.0
9                10.0                 2.0                 60.0
10               13.0                 2.0                 45.0
11               12.0                 2.0                 60.0
12               10.0                 3.0                120.0
13               10.0                 3.0                 25.0
14               12.0                 2.0                 90.0
15               12.0                 3.0                 60.0
16               10.0                 2.0                  5.0
17               12.0                 2.0                120.0
18                9.0                 2.0                 45.0
19               12.0                 2.0                 90.0
20               12.0                 1.0                  0.0
21               12.0                 1.0                240.0
22               12.0                 2.0                360.0
23               12.0                 2.0                240.0
24               12.0                 2.0                200.0
25               12.0                 3.0                360.0
26               12.0                 2.0                340.0
27               12.0                 2.0                180.0
28               10.0                 2.0                120.0
29               10.0                 1.0                 90.0
...               ...                 ...                  ...
89526             8.0                 2.0                 30.0
89536             8.0                 2.0                 20.0
```

89537	8.0	1.0	20.0
89546	8.0	1.0	0.0
89547	14.0	2.0	60.0
89549	8.0	2.0	0.0
89570	13.0	1.0	60.0
89574	6.0	2.0	15.0
89621	10.0	2.0	30.0
89630	8.0	2.0	90.0
89668	14.0	2.0	60.0
89697	0.0	2.0	0.0
89727	12.0	3.0	30.0
89778	13.0	1.0	100.0
89786	12.0	1.0	0.0
89788	14.0	2.0	0.0
89821	14.0	1.0	0.0
89852	13.0	2.0	180.0
89856	12.0	2.0	40.0
89881	10.0	3.0	90.0
89962	12.0	3.0	30.0
89969	4.0	2.0	15.0
90001	10.0	2.0	80.0
90006	0.0	2.0	120.0
90031	8.0	2.0	30.0
90065	12.0	1.0	45.0
90167	12.0	3.0	60.0
90206	8.0	2.0	30.0
90239	8.0	2.0	45.0
90327	10.0	2.0	70.0

	details.yearpublished	stats.averageweight	stats.average	stats.owned	\
0	1986.0	4.3477	7.66508	5251.0	
1	1981.0	1.9423	6.60815	1053.0	
2	1998.0	2.5085	7.44119	11870.0	
3	1992.0	2.6667	6.60675	523.0	
4	1964.0	2.5089	7.35830	18682.0	
5	1989.0	3.0000	6.52534	106.0	
6	1978.0	1.8217	6.50534	4561.0	
7	1993.0	2.4000	6.14538	460.0	
8	1998.0	3.1958	6.51776	2263.0	
9	1998.0	2.1649	6.74996	7792.0	
10	1997.0	1.6777	7.06751	39474.0	
11	1999.0	2.3560	7.47505	14413.0	
12	1995.0	2.3603	7.26569	95401.0	
13	1998.0	1.8588	6.78156	1609.0	
14	1977.0	2.3708	6.93470	3991.0	
15	1996.0	2.1538	6.84341	1027.0	
16	1999.0	1.5493	6.30870	1123.0	
17	1994.0	2.4340	7.15355	22500.0	

18	1991.0	1.8467	6.31166	2402.0
19	1988.0	3.1636	7.43592	794.0
20	1981.0	3.0000	5.35714	80.0
21	1979.0	4.4985	7.14384	3197.0
22	1979.0	3.1324	6.95654	922.0
23	1997.0	3.4902	6.66812	789.0
24	1998.0	3.2105	5.93231	622.0
25	1996.0	3.8511	7.09930	2491.0
26	1984.0	3.1404	5.57724	1941.0
27	1987.0	2.6489	6.52890	7170.0
28	1993.0	3.2857	6.66639	187.0
29	1981.0	1.8182	6.64350	1276.0
...	...	...	...	...
89526	2017.0	0.0000	6.04167	35.0
89536	2017.0	0.0000	5.69185	47.0
89537	2017.0	1.0000	6.91718	154.0
89546	2017.0	0.0000	6.74375	31.0
89547	2017.0	0.0000	7.82960	102.0
89549	2017.0	1.0000	6.37356	126.0
89570	2017.0	0.0000	7.71429	11.0
89574	2017.0	1.0000	7.15000	22.0
89621	2016.0	2.0000	7.18182	21.0
89630	2017.0	0.0000	6.78667	33.0
89668	2017.0	0.0000	8.50667	44.0
89697	2017.0	0.0000	8.45833	60.0
89727	2017.0	1.0000	7.33939	53.0
89778	2017.0	2.6667	8.21429	53.0
89786	2017.0	1.0000	7.32069	70.0
89788	2017.0	2.0000	6.46364	38.0
89821	2017.0	0.0000	9.28333	56.0
89852	2017.0	3.0000	7.53037	90.0
89856	2017.0	0.0000	8.37522	185.0
89881	2017.0	0.0000	8.45455	15.0
89962	2017.0	1.5000	6.24000	58.0
89969	2017.0	2.0000	8.67647	22.0
90001	2017.0	2.3333	6.90270	33.0
90006	2017.0	2.0000	8.09000	207.0
90031	2017.0	0.0000	6.30690	46.0
90065	2015.0	3.0000	7.59167	27.0
90167	2017.0	3.7500	5.53571	11.0
90206	2017.0	0.0000	6.30714	126.0
90239	2017.0	2.2000	7.09487	85.0
90327	2017.0	0.0000	8.26829	208.0

	stats.stddev	Abstract Strategy	Action / Dexterity	Adventure	\
0	1.593210	0	0	0	
1	1.462820	0	0	0	
2	1.185310	1	0	0	

3	1.210280	0	0	0
4	1.330200	0	0	0
5	1.650640	0	0	0
6	1.310780	1	0	0
7	1.255080	0	0	0
8	1.404130	0	0	0
9	1.246160	0	0	0
10	1.276770	0	0	0
11	1.339560	0	0	0
12	1.448420	0	0	0
13	1.209930	0	0	0
14	1.795740	0	0	0
15	1.187540	0	0	0
16	1.512460	0	0	0
17	1.516010	0	0	0
18	1.206600	0	0	0
19	1.544890	0	0	0
20	1.673620	0	0	0
21	1.958700	0	0	1
22	1.564710	0	0	0
23	1.643250	0	0	0
24	1.487550	0	0	0
25	1.750650	0	0	0
26	2.017120	0	0	0
27	1.567170	0	0	0
28	1.428250	0	0	0
29	1.790150	0	0	1
...	...	...	...	...
89526	1.540540	0	0	0
89536	1.655460	0	0	0
89537	1.166520	0	0	0
89546	1.669190	0	0	0
89547	1.283350	0	0	0
89549	1.381530	0	0	0
89570	2.736750	0	0	0
89574	0.788987	0	1	0
89621	1.748670	0	0	0
89630	1.523970	1	0	0
89668	2.165010	0	0	0
89697	1.029930	0	0	0
89727	1.483640	0	0	0
89778	2.114700	0	0	0
89786	0.882934	0	0	0
89788	1.270490	0	0	0
89821	1.737020	0	0	0
89852	3.565730	0	0	0
89856	2.084090	0	0	0
89881	1.075650	0	0	0

89962	1.443050	0	0	0
89969	1.542930	0	0	0
90001	1.112940	0	0	0
90006	1.654360	0	0	1
90031	1.010270	0	0	0
90065	0.683689	0	0	0
90167	1.505520	0	0	0
90206	1.213300	0	0	0
90239	1.115120	0	0	0
90327	1.126710	0	0	0

	Age of Reason	American Civil War	American Indian Wars	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
20	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	0	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	0	
...	...	...	...	
89526	0	0	0	
89536	0	0	0	
89537	0	0	0	
89546	0	0	0	
89547	0	0	0	

89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89697	0	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	American Revolutionary War	American West	Ancient	Animals	Arabian	\
0	0	0	0	0	0	
1	0	0	0	0	0	
2	0	0	0	0	0	
3	0	0	1	0	0	
4	0	0	0	0	0	
5	0	0	0	0	0	
6	0	0	0	0	0	
7	0	0	0	0	0	
8	0	0	0	0	0	
9	0	0	0	0	0	
10	0	0	0	0	0	
11	0	0	1	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	
16	0	0	0	0	0	
17	0	0	0	0	0	
18	0	1	0	0	0	
19	0	0	0	0	0	
20	0	0	0	0	0	

21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	0	0
28	0	0	0	0	0
29	0	0	0	0	0
...	...	...	...	...	...
89526	0	0	0	0	0
89536	0	0	0	1	0
89537	0	0	0	0	0
89546	0	0	0	0	0
89547	0	0	0	0	0
89549	0	0	0	0	0
89570	0	0	0	0	0
89574	0	0	0	0	0
89621	0	0	0	0	0
89630	0	0	0	0	0
89668	0	0	0	0	0
89697	0	0	0	0	0
89727	0	0	0	0	0
89778	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	0	0
89852	0	0	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	0	0	0	0
90001	0	0	0	0	0
90006	0	0	0	0	0
90031	0	0	0	0	0
90065	0	0	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	1	0
90327	0	0	0	0	0

	Aviation / Flight	Bluffing	Book	Card Game	Children's Game	\
0	0	0	0	0	0	
1	0	0	0	1	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	
5	0	0	0	0	0	

6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	1	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	1	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	0	0	0	0	0
18	0	1	0	0	0
19	0	0	0	0	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	1	0
28	0	0	0	0	0
29	0	0	0	0	0
...	...	...	...	...	...
89526	0	1	0	1	0
89536	0	0	0	1	0
89537	0	0	0	0	0
89546	0	0	0	1	0
89547	0	0	0	0	0
89549	0	0	0	0	0
89570	0	0	0	0	0
89574	0	0	0	0	1
89621	0	0	0	1	0
89630	0	1	0	1	0
89668	0	0	0	0	0
89697	0	0	0	0	0
89727	0	0	0	0	0
89778	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	1	0
89852	0	0	0	0	0
89856	0	0	0	1	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	0	0	1	0
90001	0	0	0	0	0

```
90006                  0         0     0         0                  0
90031                  0         1     0         1                  0
90065                  0         0     0         1                  0
90167                  0         0     0         0                  0
90206                  0         0     0         1                  0
90239                  0         0     0         0                  0
90327                  0         0     0         1                  0

       City Building  Civil War  Civilization  Collectible Components  \
0                  0          0             0                       0
1                  0          0             0                       0
2                  0          0             0                       0
3                  0          0             0                       0
4                  0          0             0                       0
5                  0          0             1                       0
6                  0          0             0                       0
7                  0          0             1                       0
8                  0          0             0                       0
9                  0          0             0                       0
10                 0          0             0                       0
11                 0          0             0                       0
12                 0          0             0                       0
13                 0          0             0                       0
14                 0          0             0                       0
15                 0          0             0                       0
16                 0          0             0                       1
17                 0          0             0                       0
18                 1          0             0                       0
19                 0          0             0                       0
20                 0          0             1                       0
21                 0          0             0                       0
22                 0          0             0                       0
23                 0          0             1                       0
24                 0          0             0                       0
25                 0          0             1                       0
26                 0          0             0                       0
27                 0          0             0                       0
28                 0          0             0                       0
29                 0          0             0                       0
...              ...        ...           ...                     ...
89526              0          0             0                       0
89536              0          0             0                       0
89537              0          0             0                       0
89546              0          0             0                       0
89547              0          0             0                       0
89549              0          0             0                       0
89570              0          0             0                       0
89574              0          0             0                       0
```

89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89697	0	0	0	0
89727	0	0	0	0
89778	1	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0
89852	0	0	1	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	1	0	0	0
90006	0	0	0	0
90031	0	0	0	0
90065	1	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Comic Book / Strip	Deduction	Dice	Economic	Educational	Electronic	\
0	0	0	0	1	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	1	0	0	
5	0	0	0	0	0	0	
6	0	0	0	0	0	0	
7	0	0	0	0	0	0	
8	0	0	0	0	0	0	
9	0	0	0	0	0	0	
10	0	0	0	0	0	0	
11	0	0	0	0	0	0	
12	0	0	0	0	0	0	
13	0	0	0	0	0	0	
14	0	0	0	0	0	0	
15	0	0	0	1	0	0	
16	0	0	1	0	0	0	
17	0	0	0	0	0	0	
18	0	0	0	0	0	0	
19	0	0	0	0	0	0	
20	0	0	0	0	0	0	
21	0	0	0	0	0	0	
22	0	0	0	0	0	0	
23	0	0	0	0	0	0	

24	0	0	0	0	0	0
25	0	0	0	1	0	0
26	0	0	0	1	0	0
27	0	0	0	0	0	0
28	0	0	0	0	0	0
29	0	0	0	0	0	1
...	...	...	...	...	...	...
89526	0	0	0	0	0	0
89536	0	0	0	0	0	0
89537	0	0	0	0	0	0
89546	0	0	0	0	0	0
89547	1	0	1	0	0	0
89549	0	0	1	0	0	0
89570	0	0	0	0	0	0
89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	0	0	0	1	0	0
89668	0	0	0	0	0	0
89697	0	0	0	0	0	0
89727	0	1	0	0	0	0
89778	0	0	1	1	0	0
89786	0	0	0	0	0	0
89788	0	1	0	0	0	0
89821	0	0	0	0	0	0
89852	0	0	0	1	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	0	0
89962	0	0	0	0	0	0
89969	0	0	1	0	0	0
90001	0	0	0	0	0	0
90006	0	0	1	0	0	0
90031	0	0	0	1	0	0
90065	0	0	0	1	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	0	0	0
90327	0	0	0	0	0	0

	Environmental	Expansion for Base-game	Exploration	Fan Expansion	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	1	0	

9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	1	0
21	0	0	1	0
22	0	0	0	0
23	0	0	0	0
24	0	0	1	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	1	0
...	...	...	...	...
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89546	0	0	0	0
89547	0	0	0	0
89549	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	1	0
89697	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0
89852	0	0	1	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90006	0	0	1	0
90031	0	0	0	0
90065	0	0	0	0

90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Fantasy	Farming	Fighting	Game System	Horror	Humor	\
0	0	0	0	0	0	0	
1	1	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	
5	0	0	0	0	0	0	
6	0	0	0	0	0	0	
7	1	0	0	0	0	0	
8	0	0	0	0	0	0	
9	1	0	0	0	0	0	
10	0	1	0	0	0	0	
11	0	0	0	0	0	0	
12	0	0	0	0	0	0	
13	0	0	0	0	0	0	
14	0	0	0	0	0	0	
15	0	0	0	0	0	0	
16	0	0	1	0	0	0	
17	0	0	0	0	0	0	
18	0	0	0	0	0	0	
19	0	0	0	0	0	0	
20	0	0	0	0	0	0	
21	1	0	0	0	0	0	
22	1	0	0	0	0	0	
23	0	0	0	0	0	0	
24	1	0	0	0	0	0	
25	0	0	0	0	0	0	
26	0	0	0	0	0	0	
27	0	0	0	0	0	1	
28	0	0	1	0	0	1	
29	1	0	1	0	0	0	
...	...	...	...	...	...	...	
89526	0	0	1	0	0	0	
89536	0	0	0	0	0	0	
89537	0	0	0	0	0	0	
89546	0	0	0	0	0	0	
89547	0	0	1	0	0	0	
89549	0	0	0	0	0	0	
89570	1	0	0	0	0	0	
89574	0	0	0	0	0	0	
89621	0	0	0	0	0	0	
89630	0	0	0	0	0	0	
89668	1	0	0	0	0	0	

89697	0	0	0	0	0	0
89727	0	0	0	0	0	0
89778	1	0	0	0	0	0
89786	0	0	0	0	0	0
89788	0	0	0	0	0	0
89821	1	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	0	0
89962	0	0	0	0	0	1
89969	0	0	0	0	0	1
90001	0	0	0	0	0	0
90006	1	0	1	0	0	0
90031	0	0	0	0	0	0
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	0	0	0
90327	1	0	0	0	0	1

	Industry / Manufacturing	Korean War	Mafia	Math	...	\
0	0	0	0	0	...	
1	0	0	0	0	...	
2	0	0	0	0	...	
3	0	0	0	0	...	
4	0	0	0	0	...	
5	0	0	0	0	...	
6	0	0	0	0	...	
7	0	0	0	0	...	
8	0	0	0	0	...	
9	0	0	0	0	...	
10	0	0	0	0	...	
11	0	0	0	0	...	
12	0	0	0	0	...	
13	0	0	0	0	...	
14	0	0	0	0	...	
15	0	0	0	0	...	
16	0	0	0	0	...	
17	0	0	0	0	...	
18	0	0	0	0	...	
19	0	0	0	0	...	
20	0	0	0	0	...	
21	0	0	0	0	...	
22	0	0	0	0	...	
23	0	0	0	0	...	
24	0	0	0	0	...	
25	0	0	0	0	...	
26	0	0	0	0	...	

27	0	0	0	0	...
28	0	0	0	0	...
29	0	0	0	0	...
...	...	...	...	...	...
89526	0	0	0	0	...
89536	0	0	0	0	...
89537	0	0	0	0	...
89546	0	0	0	0	...
89547	0	0	0	0	...
89549	0	0	0	0	...
89570	0	0	0	0	...
89574	0	0	0	0	...
89621	0	0	0	0	...
89630	1	0	0	1	...
89668	0	0	0	0	...
89697	0	0	0	0	...
89727	0	0	0	0	...
89778	0	0	0	0	...
89786	0	0	0	0	...
89788	0	0	0	0	...
89821	0	0	0	0	...
89852	0	0	0	0	...
89856	0	0	0	0	...
89881	0	0	0	0	...
89962	0	0	0	0	...
89969	0	0	0	0	...
90001	0	0	0	0	...
90006	0	0	0	0	...
90031	0	0	0	0	...
90065	0	0	0	0	...
90167	0	0	0	0	...
90206	0	0	0	0	...
90239	0	0	0	0	...
90327	0	0	0	0	...

	Action / Movement Programming	Action Point Allowance System	\
0	0	0	
1	0	0	
2	0	0	
3	0	1	
4	0	0	
5	0	0	
6	0	0	
7	0	0	
8	0	0	
9	0	0	
10	0	0	
11	0	0	

12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	1	0
18	0	0
19	0	1
20	0	0
21	1	0
22	0	0
23	0	0
24	0	0
25	0	0
26	0	0
27	0	0
28	0	1
29	0	0
...	...	...
89526	0	0
89536	0	0
89537	0	0
89546	0	0
89547	0	0
89549	0	0
89570	0	0
89574	0	0
89621	0	0
89630	0	0
89668	0	0
89697	0	0
89727	0	0
89778	0	0
89786	0	0
89788	0	0
89821	0	0
89852	0	1
89856	0	0
89881	0	0
89962	0	0
89969	0	0
90001	0	0
90006	0	0
90031	0	0
90065	0	0
90167	0	0
90206	0	0
90239	0	0

90327	0		0

	Area Control / Area Influence	Area Enclosure	Area Movement	\
0	1	0	0	
1	0	0	0	
2	1	0	0	
3	1	0	0	
4	0	0	0	
5	0	0	0	
6	0	1	0	
7	0	0	0	
8	1	0	0	
9	0	0	0	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
20	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	0	1	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	1	
...	...	...	...	
89526	0	0	0	
89536	0	0	0	
89537	0	0	0	
89546	0	0	0	
89547	0	0	0	
89549	0	0	0	
89570	1	0	0	
89574	0	0	0	
89621	0	0	0	
89630	0	0	0	
89668	0	0	0	
89697	0	0	0	
89727	0	0	0	
89778	0	0	0	

89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	1	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Area-Impulse	Auction/Bidding	Betting/Wagering	\
0	0	1	0	
1	0	0	0	
2	0	0	0	
3	0	1	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	
10	0	0	0	
11	0	1	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	1	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
20	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	1	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	0	

...	...	...	...
89526	0	0	0
89536	0	0	0
89537	0	0	0
89546	0	0	0
89547	0	0	0
89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	1	0
89668	0	0	0
89697	0	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	1
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Campaign / Battle Card Driven	Card Drafting	Chit-Pull System	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	1	0	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	

15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
20	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	1	0
28	0	0	0
29	0	0	0
...	...	...	...
89526	0	0	0
89536	0	0	0
89537	0	0	0
89546	0	0	0
89547	0	0	0
89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89697	1	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	1	0
89852	0	0	0
89856	0	0	0
89881	0	1	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	0	0
90065	0	1	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Co-operative Play	Commodity Speculation	Crayon Rail System	\

0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
20	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	1	0
27	0	0	0
28	0	0	0
29	0	0	0
...	...	...	...
89526	0	0	0
89536	0	0	0
89537	0	0	0
89546	0	0	0
89547	0	0	0
89549	0	0	0
89570	1	0	0
89574	0	0	0
89621	0	0	0
89630	0	1	0
89668	0	0	0
89697	0	0	0
89727	1	0	0
89778	0	0	0
89786	1	0	0
89788	1	0	0
89821	1	0	0

89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	1	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Deck / Pool Building	Dice Rolling	Grid Movement	Hand Management	\
0	0	1	0	1	
1	0	0	0	0	
2	0	0	0	1	
3	0	0	0	0	
4	0	0	0	1	
5	0	1	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	1	
10	0	0	0	1	
11	0	0	0	0	
12	0	1	0	1	
13	0	0	0	0	
14	0	0	0	1	
15	0	0	0	0	
16	0	1	0	0	
17	0	0	1	0	
18	0	0	0	0	
19	0	0	0	0	
20	0	0	0	0	
21	0	0	0	0	
22	0	1	0	0	
23	0	1	0	0	
24	0	0	0	0	
25	0	0	0	0	
26	0	1	0	0	
27	0	1	0	0	
28	0	0	0	0	
29	0	0	0	0	
...	...	...	...	...	
89526	0	0	0	1	
89536	0	0	0	0	

89537	0	0	0	0
89546	0	0	0	0
89547	1	1	0	0
89549	0	0	0	0
89570	0	1	0	0
89574	0	0	0	0
89621	0	0	0	1
89630	0	0	0	1
89668	0	1	0	0
89697	0	1	0	0
89727	0	0	0	0
89778	0	1	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	1	0	0	1
89852	0	1	0	1
89856	0	0	0	1
89881	0	0	0	0
89962	0	0	0	0
89969	0	1	0	0
90001	1	0	0	0
90006	0	1	1	0
90031	0	0	0	0
90065	0	0	0	1
90167	0	0	0	0
90206	0	0	0	1
90239	0	0	0	0
90327	1	0	0	1

	Hex-and-Counter	Line Drawing	Memory_mechanics	Modular Board	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	1	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	1	
13	0	0	0	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	0	1	

18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	1
22	1	0	0	0
23	1	0	0	1
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	1
29	0	0	0	0
...	...	...	...	...
89526	0	0	1	0
89536	0	0	0	0
89537	0	0	0	0
89546	0	0	0	0
89547	0	0	0	0
89549	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89697	1	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0
89852	0	0	0	1
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90006	0	0	0	1
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Paper-and-Pencil	Partnerships	Pattern Building	Pattern Recognition	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	

3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	1	1
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	0	0
...	...	...	...	...
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89546	0	0	0	0
89547	0	0	0	0
89549	0	0	0	0
89570	0	0	0	0
89574	0	0	1	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89697	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0
89852	0	0	0	0
89856	0	1	0	0
89881	0	0	0	0

89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90006	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Pick-up and Deliver	Player Elimination	Point to Point Movement	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	1	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
20	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	0	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	0	
...	...	...	...	
89526	0	0	0	
89536	0	0	0	
89537	0	0	0	
89546	0	0	0	
89547	0	0	0	

89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	1	0	0
89668	0	0	0
89697	0	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90006	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Press Your Luck	Rock-Paper-Scissors	Role Playing	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	
10	0	0	0	
11	1	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	1	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
20	0	0	0	

21	0	1	1
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	0	0
28	0	0	0
29	1	0	0
...	...	...	...
89526	0	0	0
89536	0	0	0
89537	0	0	0
89546	0	0	0
89547	0	0	0
89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89697	0	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	1	0	1
90001	0	0	0
90006	0	0	1
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Roll / Spin and Move	Route/Network Building	Secret Unit Deployment	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	

6	0	0	0
7	0	0	0
8	0	0	0
9	0	1	0
10	0	0	0
11	0	0	0
12	0	1	0
13	1	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
20	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	1	0
28	0	0	0
29	0	0	0
...	...	...	...
89526	0	0	1
89536	0	0	0
89537	0	0	0
89546	0	0	0
89547	0	0	0
89549	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89697	0	0	0
89727	0	0	0
89778	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0

90006	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Set Collection	Simulation	Simultaneous Action Selection	Singing	\
0	0	0	1	0	
1	0	0	0	0	
2	1	0	0	0	
3	1	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	1	0	0	0	
11	1	0	0	0	
12	0	0	0	0	
13	1	0	1	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	1	0	
18	0	0	0	0	
19	0	0	0	0	
20	0	0	0	0	
21	0	0	1	0	
22	0	0	0	0	
23	0	0	0	0	
24	0	0	0	0	
25	0	0	0	0	
26	0	0	0	0	
27	0	0	0	0	
28	0	0	0	0	
29	0	0	0	0	
...	...	...	...	...	
89526	0	0	0	0	
89536	0	0	0	0	
89537	0	0	1	0	
89546	0	0	0	0	
89547	0	0	0	0	
89549	0	0	0	0	
89570	0	0	0	0	
89574	0	0	0	0	

89621	1	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89697	0	1	0	0
89727	0	0	0	0
89778	1	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0
89852	0	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	1	0	0	0
90001	0	0	0	0
90006	0	0	0	0
90031	0	0	1	0
90065	0	0	0	0
90167	0	0	0	0
90206	1	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Stock Holding	Storytelling	Take That	Tile Placement	Time Track	\
0	0	0	0	0	0	
1	0	0	0	0	0	
2	0	0	0	1	0	
3	0	0	0	0	0	
4	1	0	0	1	0	
5	0	0	0	0	0	
6	0	0	0	1	0	
7	0	0	0	0	0	
8	0	0	0	1	0	
9	0	0	0	0	0	
10	0	0	0	0	0	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	
16	0	0	0	0	0	
17	0	0	0	0	0	
18	0	0	0	1	0	
19	0	0	0	0	0	
20	0	0	0	0	0	
21	0	0	0	0	0	
22	0	0	0	0	0	
23	0	0	0	1	0	

24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	1	0
28	0	0	0	0	0
29	0	0	0	0	0
...	...	...	...	...	...
89526	0	0	1	0	0
89536	0	0	0	0	0
89537	0	0	0	1	0
89546	0	0	0	0	0
89547	0	0	0	0	0
89549	0	0	0	0	0
89570	0	0	0	0	0
89574	0	0	0	0	0
89621	0	0	1	0	0
89630	1	0	1	0	0
89668	0	1	0	1	0
89697	0	0	0	0	0
89727	0	0	0	0	0
89778	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	0	0
89852	0	0	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	0	0	0	0
90001	0	0	0	0	0
90006	0	1	0	0	0
90031	0	0	0	0	0
90065	0	0	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	1	0
90327	0	0	0	0	0

	Trading	Trick-taking	Variable Phase Order	Variable Player Powers	\
0	0	0	0	0	
1	0	1	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	

9	0	0	0	0
10	1	0	0	0
11	0	0	0	0
12	1	0	0	0
13	0	0	0	0
14	0	0	0	1
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
20	0	0	0	0
21	0	0	0	1
22	0	0	1	0
23	0	0	0	1
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	1
28	0	0	0	0
29	0	0	0	0
...	...	...	...	...
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89546	0	0	0	0
89547	0	0	0	1
89549	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	1
89668	0	0	0	1
89697	0	0	0	0
89727	0	0	0	0
89778	0	0	0	1
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	1	1
89852	0	0	0	1
89856	0	0	0	0
89881	0	0	0	1
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90006	0	0	0	1
90031	0	0	0	0
90065	0	0	0	0

90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Voting	Worker Placement
0	0	0
1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	1	0
19	0	0
20	0	0
21	0	0
22	0	0
23	1	0
24	0	0
25	0	0
26	0	0
27	0	0
28	0	0
29	0	0
...	...	...
89526	0	0
89536	0	0
89537	0	0
89546	0	0
89547	0	0
89549	0	0
89570	0	0
89574	0	0
89621	0	0
89630	0	0
89668	0	0

```
89697       0                0
89727       0                0
89778       0                1
89786       0                0
89788       0                0
89821       0                0
89852       0                0
89856       1                0
89881       0                0
89962       0                0
89969       0                0
90001       0                0
90006       0                0
90031       0                0
90065       0                1
90167       0                0
90206       0                0
90239       0                0
90327       0                0

[20842 rows x 145 columns]
```

```
In [24]: len(df_cats.columns) + len(df_mechs.columns) + len(df_limit.columns) - len(df_data.co
```

```
Out[24]: True
```

```
In [25]: #Check if there are any NaNs
         nanlist = df_data.isnull().sum(axis=1).tolist()

         for i in nanlist:
             if i != 0:
                 print("Boop")
         print("done")
```

```
done
```

```
In [26]: #We see that there are some games that have the same name, but not that many.
         df_data['details.name'].value_counts()
```

```
Out[26]: Robin Hood                                    5
         Mafia                                        5
         Cosmic Encounter                             4
         Gangster                                     4
         Quicksand                                    4
         Chaos                                        4
         Samurai                                      4
         Saga                                         4
         Grand Prix                                   4
```

Vegas	4
Battle of the Bulge	3
Around the World in 80 Days	3
Versailles	3
Combat	3
Polarity	3
Artifact	3
Dallas	3
The Hobbit: An Unexpected Journey	3
Horus Heresy	3
Austerlitz	3
En Garde!	3
Waterloo	3
Blindes Huhn	3
Guadalcanal	3
Inferno	3
King Arthur	3
Barbarossa	3
Siege	3
Witch Hunt	3
Imperator	3
	..
Creationary	1
Just a Peiper Dream?	1
I Say, Holmes! (Second Edition)	1
Hooker and Lee	1
Emerald	1
Mentalis	1
Piraten	1
Battlestar Galactica Collectible Card Game	1
Strafexpedition 1916	1
Red Army	1
Congo Merc: The Congo, 1964	1
Hero of Weehawken	1
A Famous Victory	1
Rincala	1
Wabbit Wampage	1
How the Grinch Stole Christmas! Game	1
Rails Through the Rockies	1
Strategos	1
Wacky Pirates	1
Maestro	1
Cauldron Quest	1
Soldiers: Man-to-Man Combat in World War II	1
Festival of Thousand Cats	1
Uncharted: The Board Game	1
Mount Everest	1
Breaking Away	1

```
Panzer Grenadier: Kursk  Burning Tigers       1
1000 Bornes Express                            1
Edison & Co.                                   1
Antartik                                       1
Name: details.name, Length: 20340, dtype: int64
```

1.3.3 Processing Response Variables

The output variable for this analysis will be "success", as measured by the player enjoyment rating (using a 10-star likert scale). The variables listed below capture this outcome in various ways.

Response Variables

- stats.average (0-10 stars)
- stats.stddev (for average star rating)

We'll make two adjustments to our outcome variable. - As given, the stats.average variable is a continuous variable between 0 and 10. Our models will work best with a finite number of outcome groups, so we'll bin the rating by star levels. To keep things simple, we'll round down each star rating to a whole number (so 5.7 will become 5, for example). This will be stored as "binned_average". - Some of our models may work best with a binary outcome measurement. For that, we can create a binary measure for the player enjoyment rating. Games with an average rating greater than 7 are considered successful, and those less than 7 are considered unsuccessful. (This threshold is set based on our experience, and has no concrete mathematical derivation.) We'll store these values as the integers 1 (success) and 0 (unsuccessful).

```
In [27]: df_data['binned.average'] = np.floor(df_data['stats.average'])
         df_data['binary.success'] = (df_data['binned.average'] > 6.5)*1
```

1.4 Exporatory Data Analaysis

With our new data frame, we can conduct an EDA to begin to look at the data. This will help us perform any additional filtering or processing that is necessary; it may also help reveal insights into game trends.

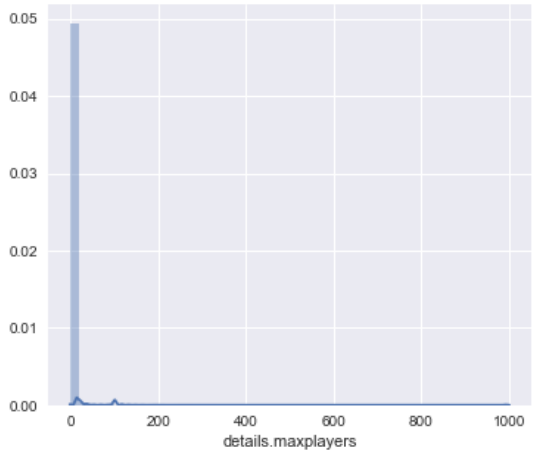
The cells below explore the key variables in our dataset. We also investigate apparent outliers, to see if they should be removed from the analysis.

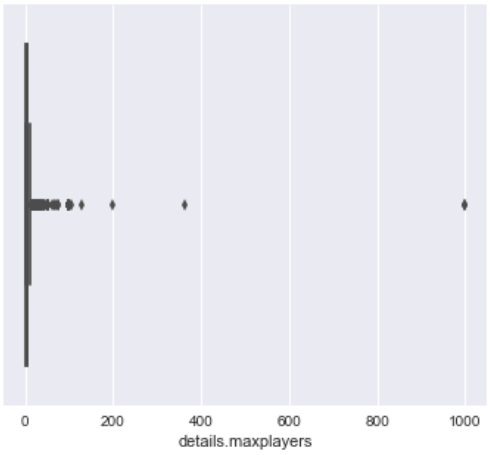
```
In [28]: import matplotlib.pyplot as plt
         def analyze_feature(data, feat_name):
             print(data[feat_name].describe())
             plt.figure(figsize = (20,5))
             plt.subplot(1,3,2)
             sns.distplot(data[feat_name])
             plt.subplot(1,3,3)
             sns.boxplot(data[feat_name])
```

MaxPlayers

```
In [29]: analyze_feature(df_data, "details.maxplayers")
```

```
count    20842.000000
mean         5.341618
std         14.373976
min          0.000000
25%          2.000000
50%          4.000000
75%          6.000000
max        999.000000
Name: details.maxplayers, dtype: float64
```





In [30]:
```
df_data.loc[df_data['details.maxplayers']==999]
```

Out[30]:

	details.name	details.maxplayers	details.minage	details.minplayers	details.playingtime	details.yearpublished	stats.averageweight	stats.average	stats.owned	stats.stddev	Abstract Strategy	Action / Dexterity	Adventure
23008	Start Player: A Kinda Collectible Card Game	999.0	6.0	2.0	1.0	2006.0	1.0000	6.49793	263.0	1.91599	0	0	0
26326	I Don't Know, What Do You Want To Play?	999.0	8.0	2.0	5.0	2007.0	1.0625	6.81325	117.0	1.89398	0	0	0
81044	Scrimish Card Game	999.0	8.0	2.0	100.0	2015.0	1.3333	6.13548	297.0	1.60829	0	0	0

```
       Age of Reason  American Civil War  American Indian Wars  \
23008              0                   0                     0
26326              0                   0                     0
81044              0                   0                     0

       American Revolutionary War  American West  Ancient  Animals  Arabian  \
23008                           0              0        0        0        0
26326                           0              0        0        0        0
81044                           0              0        0        0        0

       Aviation / Flight  Bluffing  Book  Card Game  Children's Game  \
23008                  0         0     0          1                0
26326                  0         0     0          1                0
81044                  0         0     0          1                0

       City Building  Civil War  Civilization  Collectible Components  \
23008              0          0             0                       1
26326              0          0             0                       0
81044              0          0             0                       0

       Comic Book / Strip  Deduction  Dice  Economic  Educational  Electronic  \
23008                   1          0     0         0            0           0
26326                   0          0     0         0            0           0
81044                   0          0     0         0            0           0

       Environmental  Expansion for Base-game  Exploration  Fan Expansion  \
23008              0                        0            0              0
26326              0                        0            0              0
81044              0                        0            0              0

       Fantasy  Farming  Fighting  Game System  Horror  Humor  \
23008        0        0         0            0       0      1
26326        0        0         0            0       0      0
81044        0        0         0            0       0      0

       Industry / Manufacturing  Korean War  Mafia  Math       ...        \
23008                         0           0      0     0       ...
26326                         0           0      0     0       ...
81044                         0           0      0     0       ...

       Area Control / Area Influence  Area Enclosure  Area Movement  \
23008                              0               0              0
26326                              0               0              0
81044                              0               0              0

       Area-Impulse  Auction/Bidding  Betting/Wagering  \
23008             0                0                 0
```

```
26326            0               0                0
81044            0               0                0

       Campaign / Battle Card Driven  Card Drafting  Chit-Pull System  \
23008                              0              0                 0
26326                              0              0                 0
81044                              0              0                 0

       Co-operative Play  Commodity Speculation  Crayon Rail System  \
23008                  0                      0                   0
26326                  0                      0                   0
81044                  0                      0                   0

       Deck / Pool Building  Dice Rolling  Grid Movement  Hand Management  \
23008                     0             0              0                0
26326                     0             0              0                0
81044                     0             0              0                0

       Hex-and-Counter  Line Drawing  Memory_mechanics  Modular Board  \
23008                0             0                 0              0
26326                0             0                 0              0
81044                0             0                 1              0

       Paper-and-Pencil  Partnerships  Pattern Building  Pattern Recognition  \
23008                 0             0                 0                    0
26326                 0             0                 0                    0
81044                 0             0                 0                    0

       Pick-up and Deliver  Player Elimination  Point to Point Movement  \
23008                    0                   0                        0
26326                    0                   0                        0
81044                    0                   0                        0

       Press Your Luck  Rock-Paper-Scissors  Role Playing  \
23008                0                   0             0
26326                0                   0             0
81044                0                   0             0

       Roll / Spin and Move  Route/Network Building  Secret Unit Deployment  \
23008                     0                       0                       0
26326                     0                       0                       0
81044                     0                       0                       0

       Set Collection  Simulation  Simultaneous Action Selection  Singing  \
23008               0           0                              0        0
26326               0           0                              0        0
81044               0           0                              0        0
```

```
       Stock Holding  Storytelling  Take That  Tile Placement  Time Track  \
23008              0             0          0               0           0
26326              0             0          0               0           0
81044              0             0          0               0           0

       Trading  Trick-taking  Variable Phase Order  Variable Player Powers  \
23008        0             0                     0                       0
26326        0             0                     0                       0
81044        0             0                     0                       0

       Voting  Worker Placement  binned.average  binary.success
23008       0                 0             6.0               0
26326       1                 0             6.0               0
81044       0                 0             6.0               0

[3 rows x 147 columns]
```

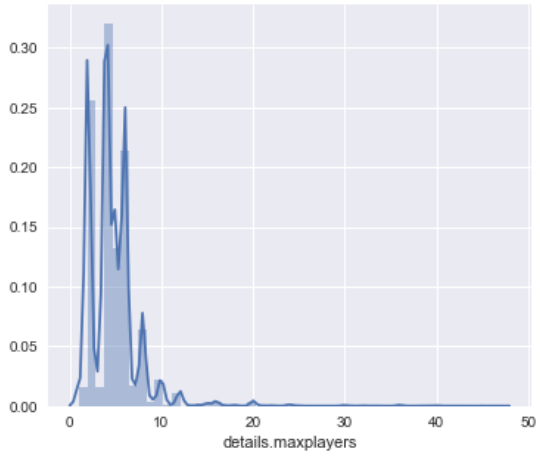
We see from the above analysis, that there are some games that could be considered outliers, as they have values for the maximum number of players, approaching 999 - which in itself might be an artificial contraint that the source imposes on game data input. Consequently it is the judgement of the investigators to limit the games investigated to a measure of maximum players three positive and negative standard deviations around the mean.

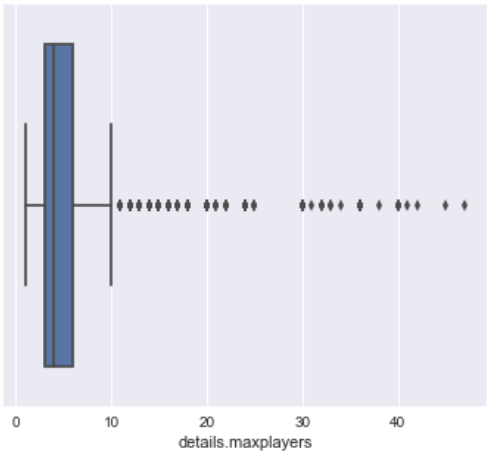
```
In [31]: df_data['details.maxplayers'].mean()

Out[31]: 5.341617886959025

In [32]: df_data['details.maxplayers'].std()

Out[32]: 14.373975920228531

In [33]: df_data['details.maxplayers'].std()*3

Out[33]: 43.121927760685594

In [34]: df_data['details.maxplayers'].mean()+df_data['details.maxplayers'].std()*3

Out[34]: 48.46354564764462
```

We see that the mean number of players is approximately 5. Consequently three standard deviations below this value is a negative number that does not make sense. Neither does a board game that can be played with zero players. Consequently we will set our minimum limit to 1 player. Our maximum player limit will be 48.46, or rounded to 49 players.

```
In [35]: df_data1 = df_data[(df_data['details.maxplayers']>=1) & (df_data['details.maxplayers']

In [36]: analyze_feature(df_data1, "details.maxplayers")
```

```
count    20367.000000
mean         4.730348
std          2.948834
min          1.000000
25%          3.000000
50%          4.000000
75%          6.000000
max         47.000000
Name: details.maxplayers, dtype: float64
```






```
In [37]: df_data['details.maxplayers'].count()-df_data1['details.maxplayers'].count()
```

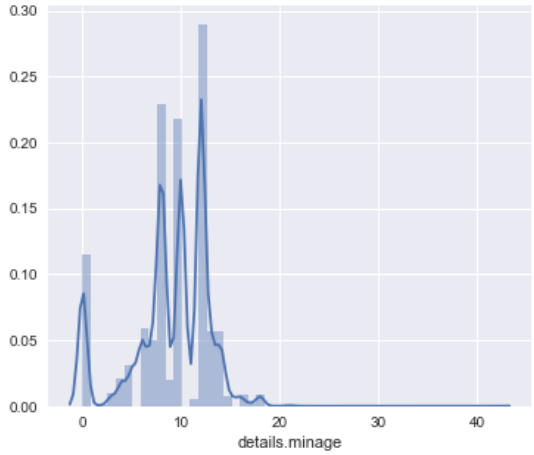
```
Out[37]: 475
```

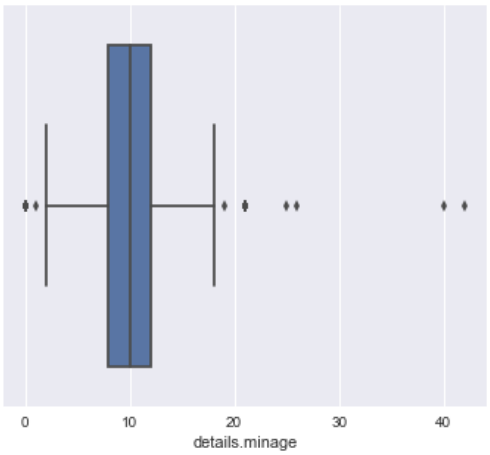
While we still observe a number of outliers based on the box and whisker plot, the investigators believe that this proceedure above will increase the robustness of our analysis. We observe we have filtered out 475 records.

MinAge

```
In [38]: analyze_feature(df_data1, "details.minage")
```

```
count    20367.000000
mean         8.989198
std          3.959902
min          0.000000
25%          8.000000
50%         10.000000
75%         12.000000
max         42.000000
Name: details.minage, dtype: float64
```





In [39]:
```
df_data.loc[df_data['details.minage']==0]
```

Out[39]:

	details.name	details.maxplayers
100	Peaceful Resistance	3.0
133	Wortelboer	4.0
152	Black Death	6.0
221	Politika	7.0
258	Rossyïa 1917	5.0
337	Wrott & Swindlers	6.0
402	Beutelschneider	4.0
430	Age of Arguments	4.0
447	Sufferin' Spirits	4.0
504	Doolittle & Waite	15.0
508	Quartier Latin	6.0
511	Terrain Game	6.0
526	Sumera	4.0
551	Cube Fusion	2.0
571	Blindside	4.0
575	Oodles	10.0
597	Plague & Pestilence	6.0
611	Marrakesh	2.0
619	Fibonacci	2.0
629	Caesar in Gallia	2.0
644	Oregon Trail	8.0
650	The Sun Never Sets	2.0
668	Dutch Mountains	2.0
768	Atlantis	0.0
774	Pythago	6.0
780	Derby	5.0
899	Xenophon: 10,000 Against Persia	2.0
1015	Tendix	3.0

```
1020                                        Casablanca                4.0
1063                                  Saludos Amigos!                 7.0
...                                               ...                ...
84695  Nine Years: The War of the Grand Alliance 1688...                2.0
84698           Buffy the Vampire Slayer: The Board Game                6.0
84739                                       Gorechosen                4.0
84827                                The Shooting Party               1.0
84828                                        Ta-Da!                   6.0
84843              Evil Dead 2: The Official Board Game               6.0
84871                                          Akua                   4.0
84952                                       Handsket                  6.0
85734                            Shadow War: Armageddon               2.0
86820                                       Xenofera                  5.0
86842                                SUPERHOT Card Game               3.0
86845                                         Gangsi                  5.0
86862                              Wallenstein Big Box                5.0
87041                                   Iberian Rails                 5.0
87139                               Thunder in the Ozarks             2.0
87297                                 Ahead in the Clouds             2.0
87363                                          Siege                  6.0
87557                                Phobos Rising!                   1.0
87762              Gwent: Nilfgaard and Northern Realms               2.0
87911                           Fall Of The Third Reich               4.0
88111                                Dia de los Muertos               6.0
88240              The Horus Heresy: Burning of Prospero              2.0
88378                           Omen: Edge of the Aegean               2.0
88571                                  Circle the Wagons               2.0
88733                                      Pyramid Poker               2.0
88797                                        Tenkatoitsu               2.0
89057                     Tao Long: The Way of the Dragon              2.0
89060                     Ascension: Gift of the Elements              4.0
89697                                  Napoleons Quagmire              2.0
90006              Warhammer Quest: Shadows Over Hammerhal             5.0

       details.minage  details.minplayers  details.playingtime  \
100               0.0                 2.0                 30.0
133               0.0                 2.0                 30.0
152               0.0                 2.0                120.0
221               0.0                 3.0                  0.0
258               0.0                 2.0                300.0
337               0.0                 3.0                 90.0
402               0.0                 3.0                 30.0
430               0.0                 4.0                120.0
447               0.0                 2.0                120.0
504               0.0                 4.0                 90.0
508               0.0                 4.0                 45.0
511               0.0                 2.0                 60.0
526               0.0                 2.0                 30.0
```

551	0.0	2.0	15.0
571	0.0	4.0	0.0
575	0.0	3.0	0.0
597	0.0	2.0	30.0
611	0.0	2.0	45.0
619	0.0	2.0	30.0
629	0.0	2.0	300.0
644	0.0	1.0	120.0
650	0.0	2.0	180.0
668	0.0	2.0	0.0
768	0.0	0.0	0.0
774	0.0	1.0	60.0
780	0.0	3.0	60.0
899	0.0	2.0	120.0
1015	0.0	2.0	45.0
1020	0.0	2.0	30.0
1063	0.0	4.0	60.0
...	...	...	...
84695	0.0	2.0	0.0
84698	0.0	1.0	60.0
84739	0.0	2.0	75.0
84827	0.0	1.0	15.0
84828	0.0	2.0	0.0
84843	0.0	1.0	90.0
84871	0.0	2.0	40.0
84952	0.0	1.0	30.0
85734	0.0	2.0	0.0
86820	0.0	2.0	75.0
86842	0.0	1.0	0.0
86845	0.0	2.0	0.0
86862	0.0	3.0	120.0
87041	0.0	2.0	90.0
87139	0.0	2.0	0.0
87297	0.0	2.0	20.0
87363	0.0	2.0	0.0
87557	0.0	1.0	0.0
87762	0.0	2.0	30.0
87911	0.0	1.0	600.0
88111	0.0	3.0	15.0
88240	0.0	2.0	90.0
88378	0.0	2.0	30.0
88571	0.0	2.0	15.0
88733	0.0	2.0	0.0
88797	0.0	1.0	0.0
89057	0.0	2.0	30.0
89060	0.0	1.0	60.0
89697	0.0	2.0	0.0
90006	0.0	2.0	120.0

	details.yearpublished	stats.averageweight	stats.average	stats.owned	\
100	1995.0	0.0000	5.62778	44.0	
133	1999.0	0.0000	4.11765	37.0	
152	1993.0	1.8421	5.70588	244.0	
221	1996.0	1.6667	4.48936	78.0	
258	1995.0	4.0000	7.25508	189.0	
337	1995.0	1.8000	6.48438	57.0	
402	1997.0	1.5000	6.03429	95.0	
430	1998.0	0.0000	5.57143	27.0	
447	1987.0	2.0000	5.86842	53.0	
504	1986.0	1.8571	6.21793	149.0	
508	1997.0	0.0000	5.18750	56.0	
511	1997.0	1.6667	4.92857	13.0	
526	1999.0	3.0000	6.31071	63.0	
551	1968.0	2.0000	5.33333	71.0	
571	1998.0	2.0000	6.25294	36.0	
575	1992.0	1.3684	6.15255	481.0	
597	1993.0	1.3889	6.12509	781.0	
611	1978.0	2.0000	7.03191	79.0	
619	1992.0	1.8333	6.33246	112.0	
629	1993.0	3.0714	6.68922	319.0	
644	1981.0	2.2500	5.48649	108.0	
650	1997.0	2.8571	6.91463	162.0	
668	2000.0	1.0000	5.16667	79.0	
768	2000.0	0.0000	6.27273	25.0	
774	2000.0	1.0000	4.31818	28.0	
780	2000.0	1.6667	6.12237	69.0	
899	2000.0	3.7500	6.49600	181.0	
1015	1998.0	2.0000	5.67857	39.0	
1020	1977.0	1.0000	4.96875	42.0	
1063	1996.0	1.8750	5.45704	141.0	
...	...	...	...	...	
84695	2017.0	0.0000	8.57308	63.0	
84698	2016.0	2.6000	7.31916	597.0	
84739	2016.0	2.0000	7.39058	372.0	
84827	2016.0	1.5000	6.77273	31.0	
84828	2016.0	1.0000	6.35795	193.0	
84843	2016.0	2.0000	8.46909	74.0	
84871	2016.0	0.0000	6.31452	246.0	
84952	2016.0	1.0000	7.35385	14.0	
85734	2017.0	4.0000	8.64286	131.0	
86820	2017.0	0.0000	7.34375	10.0	
86842	2017.0	2.0000	7.07857	90.0	
86845	2016.0	1.0000	6.14919	55.0	
86862	2016.0	3.0000	7.92550	189.0	
87041	2017.0	0.0000	7.66667	10.0	
87139	2016.0	3.5000	7.91667	104.0	

87297	2016.0	1.5000	7.51923	203.0
87363	2016.0	1.5000	5.50000	70.0
87557	2016.0	2.0000	6.80000	118.0
87762	2016.0	1.5000	7.10388	254.0
87911	2016.0	0.0000	8.00000	37.0
88111	2016.0	0.0000	6.06000	55.0
88240	2016.0	0.0000	7.73714	198.0
88378	2016.0	2.0000	7.95444	340.0
88571	2017.0	2.0000	7.95455	35.0
88733	2017.0	2.0000	7.20690	67.0
88797	2016.0	2.4000	8.62000	98.0
89057	2017.0	2.6667	8.55355	142.0
89060	2017.0	2.5000	7.82174	95.0
89697	2017.0	0.0000	8.45833	60.0
90006	2017.0	2.0000	8.09000	207.0

	stats.stddev	Abstract Strategy	Action / Dexterity	Adventure	\
100	1.570490	1	0	0	
133	1.996540	0	0	0	
152	1.606720	0	0	0	
221	1.772990	0	0	0	
258	1.280400	0	0	0	
337	1.118510	0	0	0	
402	1.376320	0	0	0	
430	1.678190	0	0	0	
447	1.435290	0	0	0	
504	1.240740	0	0	0	
508	1.058820	0	0	0	
511	1.656770	0	0	0	
526	1.723650	1	0	0	
551	1.247220	1	0	0	
571	0.994170	0	0	0	
575	1.547240	0	0	0	
597	1.655740	0	0	0	
611	1.955360	1	0	0	
619	1.466300	1	0	0	
629	1.280140	0	0	0	
644	1.449540	0	0	1	
650	1.456280	0	0	0	
668	1.126870	1	0	0	
768	0.862439	0	0	0	
774	1.418590	0	0	0	
780	1.096690	0	0	0	
899	1.562680	0	0	0	
1015	1.357610	0	0	0	
1020	1.152020	0	0	0	
1063	1.443400	0	0	0	
...	...	...	...	...	

84695	0.634464	0	0	0
84698	1.523260	0	0	0
84739	1.299980	0	0	0
84827	1.346100	0	0	0
84828	1.769100	0	0	0
84843	2.701600	0	0	0
84871	1.942690	0	0	0
84952	1.145300	0	1	0
85734	0.921216	0	0	0
86820	2.590360	0	0	0
86842	2.001490	0	0	0
86845	1.292050	0	0	0
86862	1.443070	0	0	0
87041	2.645750	0	0	0
87139	2.191210	0	0	0
87297	1.226560	0	0	0
87363	2.107810	0	0	0
87557	0.806226	0	0	0
87762	1.641910	0	0	0
87911	2.287090	0	0	0
88111	1.756060	0	0	0
88240	1.269660	0	0	0
88378	1.426250	0	0	0
88571	1.096580	0	0	0
88733	1.477400	0	0	0
88797	0.930376	0	0	0
89057	1.223250	1	0	0
89060	2.156070	0	0	0
89697	1.029930	0	0	0
90006	1.654360	0	0	1

	Age of Reason	American Civil War	American Indian Wars	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	1	0	0	
447	0	0	0	
504	0	0	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	

597	0	0	0
611	0	0	0
619	0	0	0
629	0	0	0
644	0	0	0
650	0	0	0
668	0	0	0
768	0	0	0
774	0	0	0
780	0	0	0
899	0	0	0
1015	0	0	0
1020	0	0	0
1063	0	0	0
...	...	...	...
84695	0	0	0
84698	0	0	0
84739	0	0	0
84827	0	0	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	0
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	0	0
87139	0	1	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	0

	American Revolutionary War	American West	Ancient	Animals	Arabian	\
100	0	0	0	0	0	

133	0	0	0	1	0
152	0	0	0	0	0
221	0	0	0	0	0
258	0	0	0	0	0
337	0	0	0	0	0
402	0	0	0	0	0
430	0	0	0	0	0
447	0	0	0	0	0
504	0	0	0	0	0
508	0	0	0	0	0
511	0	0	0	0	0
526	0	0	0	0	0
551	0	0	0	0	0
571	0	0	0	0	0
575	0	0	0	0	0
597	0	0	0	0	0
611	0	0	0	0	0
619	0	0	0	0	0
629	0	0	1	0	0
644	0	1	0	0	0
650	0	0	0	0	0
668	0	0	0	0	0
768	0	0	0	0	0
774	0	0	0	0	0
780	0	0	0	0	0
899	0	0	1	0	0
1015	0	0	0	0	0
1020	0	0	0	0	0
1063	0	0	0	0	0
...	...	...	...	...	...
84695	0	0	0	0	0
84698	0	0	0	0	0
84739	0	0	0	0	0
84827	0	0	0	0	0
84828	0	0	0	0	0
84843	0	0	0	0	0
84871	0	0	0	0	0
84952	0	0	0	0	0
85734	0	0	0	0	0
86820	0	0	0	0	0
86842	0	0	0	0	0
86845	0	0	0	0	0
86862	0	0	0	0	0
87041	0	0	0	0	0
87139	0	0	0	0	0
87297	0	0	0	0	0
87363	0	0	0	0	0
87557	0	0	0	0	0

87762	0	0	0	0	0
87911	0	0	0	0	0
88111	0	0	0	0	0
88240	0	0	0	0	0
88378	0	0	1	0	0
88571	0	1	0	0	0
88733	0	0	0	0	0
88797	0	0	0	0	0
89057	0	0	0	0	0
89060	0	0	0	0	0
89697	0	0	0	0	0
90006	0	0	0	0	0

	Aviation / Flight	Bluffing	Book	Card Game	Children's Game	\
100	0	0	0	0	0	
133	0	0	0	1	0	
152	0	0	0	0	0	
221	0	0	0	1	0	
258	0	0	0	0	0	
337	0	1	0	1	0	
402	0	0	0	1	0	
430	0	0	0	0	0	
447	0	0	0	0	0	
504	0	1	0	1	0	
508	0	0	0	1	0	
511	0	0	0	0	0	
526	0	0	0	0	0	
551	0	0	0	0	0	
571	0	0	0	1	0	
575	0	0	0	0	0	
597	0	0	0	1	0	
611	0	0	0	1	0	
619	0	0	0	0	0	
629	0	0	0	0	0	
644	0	0	0	0	0	
650	0	0	0	0	0	
668	0	0	0	0	0	
768	0	0	0	0	0	
774	0	0	0	0	0	
780	0	1	0	0	0	
899	0	0	0	0	0	
1015	0	0	0	0	0	
1020	0	1	0	0	0	
1063	0	0	0	0	0	
...	...	...	...	...	...	
84695	0	0	0	0	0	
84698	0	0	0	0	0	
84739	0	0	0	0	0	

84827	0	0	0	1	0
84828	0	0	0	0	0
84843	0	0	0	0	0
84871	0	0	0	0	0
84952	0	0	0	0	0
85734	0	0	0	0	0
86820	0	0	0	1	0
86842	0	0	0	1	0
86845	0	0	0	0	1
86862	0	0	0	0	0
87041	0	0	0	0	0
87139	0	0	0	0	0
87297	0	0	0	1	0
87363	0	1	0	1	0
87557	0	0	0	0	0
87762	0	0	0	1	0
87911	0	0	0	0	0
88111	0	0	0	0	0
88240	0	0	0	0	0
88378	0	0	0	1	0
88571	0	0	0	1	0
88733	0	0	0	0	0
88797	0	0	0	0	0
89057	0	0	0	0	0
89060	0	0	0	1	0
89697	0	0	0	0	0
90006	0	0	0	0	0

	City Building	Civil War	Civilization	Collectible Components	\
100	0	0	0	0	
133	0	0	0	0	
152	0	0	0	0	
221	0	0	0	0	
258	0	1	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	0	0	0	
511	0	0	0	0	
526	0	0	0	0	
551	0	0	0	0	
571	0	0	0	0	
575	0	0	0	0	
597	0	0	0	0	
611	0	0	0	0	
619	0	0	0	0	

629	0	0	0	0
644	0	0	0	0
650	0	0	0	0
668	0	0	0	0
768	0	0	0	0
774	0	0	0	0
780	0	0	0	0
899	0	0	0	0
1015	0	0	0	0
1020	0	0	0	0
1063	0	0	0	0
...	...	...	...	...
84695	0	0	0	0
84698	0	0	0	0
84739	0	0	0	0
84827	0	0	0	0
84828	0	0	0	0
84843	0	0	0	0
84871	0	0	0	0
84952	0	0	0	0
85734	0	0	0	0
86820	0	0	0	0
86842	0	0	0	0
86845	0	0	0	0
86862	0	0	0	0
87041	0	0	0	0
87139	0	0	0	0
87297	0	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	1
87911	0	0	0	0
88111	0	0	0	0
88240	0	0	0	0
88378	0	0	0	0
88571	1	0	0	0
88733	0	0	0	0
88797	0	0	0	0
89057	0	0	0	0
89060	0	0	0	0
89697	0	0	0	0
90006	0	0	0	0

	Comic Book / Strip	Deduction	Dice	Economic	Educational	Electronic	\
100	0	0	0	0	0	0	
133	0	0	0	0	0	0	
152	0	0	0	0	1	0	
221	0	0	0	0	0	0	

258	0	0	0	0	0	0
337	0	0	0	0	0	0
402	0	0	0	0	0	0
430	0	0	0	0	0	0
447	0	0	0	0	0	0
504	0	0	0	0	0	0
508	0	0	0	0	0	0
511	0	0	0	0	0	0
526	0	0	0	0	0	0
551	0	0	0	0	0	0
571	0	0	0	0	0	0
575	0	0	0	0	0	0
597	0	0	0	0	0	0
611	0	0	1	0	0	0
619	0	0	0	0	0	0
629	0	0	0	0	0	0
644	0	0	0	0	0	0
650	0	0	0	0	0	0
668	0	0	0	0	0	0
768	0	0	0	0	0	0
774	0	0	0	0	0	0
780	0	0	0	0	0	0
899	0	0	0	0	0	0
1015	0	0	0	0	0	0
1020	0	0	0	0	0	0
1063	0	0	0	0	0	0
...	...	...	...	...	...	...
84695	0	0	0	0	0	0
84698	0	0	0	0	0	0
84739	0	0	1	0	0	0
84827	0	0	0	0	0	0
84828	0	0	0	0	0	0
84843	0	0	0	0	0	0
84871	0	0	0	0	0	0
84952	0	0	0	0	0	0
85734	0	0	0	0	0	0
86820	0	0	0	0	0	0
86842	0	0	0	0	0	0
86845	0	1	0	0	0	0
86862	0	0	0	1	0	0
87041	0	0	0	1	0	0
87139	0	0	0	0	0	0
87297	0	0	0	0	0	0
87363	0	0	0	0	0	0
87557	0	0	0	0	0	0
87762	0	0	0	0	0	0
87911	0	0	0	0	0	0
88111	0	0	0	0	0	0

88240	0	0	0	0	0	0
88378	0	0	0	0	0	0
88571	0	0	0	0	0	0
88733	0	0	0	0	0	0
88797	0	0	0	0	0	0
89057	0	0	0	0	0	0
89060	0	0	0	0	0	0
89697	0	0	0	0	0	0
90006	0	0	1	0	0	0

	Environmental	Expansion for Base-game	Exploration	Fan Expansion	\
100	0	0	0	0	
133	0	0	0	0	
152	0	0	0	0	
221	0	0	0	0	
258	0	0	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	0	0	0	
511	0	0	0	0	
526	0	0	0	0	
551	0	0	0	0	
571	0	0	0	0	
575	0	0	0	0	
597	0	0	0	0	
611	0	0	0	0	
619	0	0	0	0	
629	0	0	0	0	
644	0	0	1	0	
650	0	0	0	0	
668	0	0	0	0	
768	0	0	0	0	
774	0	0	0	0	
780	0	0	0	0	
899	0	0	0	0	
1015	0	0	0	0	
1020	0	0	0	0	
1063	0	0	0	0	
...	...	...	...	...	
84695	0	0	0	0	
84698	0	0	0	0	
84739	0	0	0	0	
84827	0	0	0	0	
84828	0	0	0	0	
84843	0	0	0	0	

84871	0	0	0	0
84952	0	0	0	0
85734	0	0	0	0
86820	0	0	0	0
86842	0	0	0	0
86845	0	0	0	0
86862	0	0	0	0
87041	0	0	0	0
87139	0	0	0	0
87297	0	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	0
87911	0	0	0	0
88111	0	0	0	0
88240	0	0	0	0
88378	0	0	0	0
88571	0	0	0	0
88733	0	0	0	0
88797	0	0	0	0
89057	0	0	0	0
89060	0	0	0	0
89697	0	0	0	0
90006	0	0	1	0

	Fantasy	Farming	Fighting	Game System	Horror	Humor	\
100	0	0	0	0	0	0	
133	0	0	0	0	0	0	
152	0	0	0	0	0	0	
221	0	0	0	0	0	0	
258	0	0	0	0	0	0	
337	0	0	0	0	0	0	
402	0	0	0	0	0	0	
430	0	0	0	0	0	0	
447	0	0	0	0	1	0	
504	0	0	0	0	0	0	
508	0	0	0	0	0	0	
511	0	0	0	0	0	0	
526	0	0	0	0	0	0	
551	0	0	0	0	0	0	
571	0	0	0	0	0	0	
575	0	0	0	0	0	0	
597	0	0	0	0	0	0	
611	0	0	0	0	0	0	
619	0	0	0	0	0	0	
629	0	0	0	0	0	0	
644	0	0	0	0	0	0	
650	0	0	0	0	0	0	

668	0	0	0	0	0	0
768	0	0	0	0	0	0
774	0	0	0	0	0	0
780	0	0	0	0	0	0
899	0	0	0	0	0	0
1015	0	0	0	0	0	0
1020	0	0	0	0	0	0
1063	0	0	0	0	0	0
...	...	...	...	...	...	...
84695	0	0	0	0	0	0
84698	0	0	0	0	0	0
84739	1	0	1	0	0	0
84827	0	0	0	0	0	0
84828	0	0	0	0	0	0
84843	0	0	0	0	1	0
84871	0	0	0	0	0	0
84952	0	0	0	0	0	0
85734	0	0	0	0	0	0
86820	0	0	0	0	0	0
86842	0	0	1	0	0	0
86845	0	0	0	0	0	0
86862	0	0	0	0	0	0
87041	0	0	0	0	0	0
87139	0	0	0	0	0	0
87297	0	0	0	0	0	0
87363	0	0	1	0	0	0
87557	0	0	0	0	0	0
87762	0	0	0	0	0	0
87911	0	0	0	0	0	0
88111	0	0	0	0	1	0
88240	0	0	1	0	0	0
88378	1	0	1	0	0	0
88571	0	0	0	0	0	0
88733	0	0	0	0	0	0
88797	0	0	0	0	0	0
89057	0	0	0	0	0	0
89060	1	0	0	0	0	0
89697	0	0	0	0	0	0
90006	1	0	1	0	0	0

	Industry / Manufacturing	Korean War	Mafia	Math	...	\
100	0	0	0	0	...	
133	0	0	0	0	...	
152	0	0	0	0	...	
221	0	0	0	0	...	
258	0	0	0	0	...	
337	0	0	0	0	...	
402	0	0	0	0	...	

430	0	0	0	0	...
447	0	0	0	0	...
504	0	0	0	0	...
508	0	0	0	0	...
511	0	0	0	0	...
526	0	0	0	0	...
551	0	0	0	0	...
571	0	0	0	0	...
575	0	0	0	0	...
597	0	0	0	0	...
611	0	0	0	0	...
619	0	0	0	0	...
629	0	0	0	0	...
644	0	0	0	0	...
650	0	0	0	0	...
668	0	0	0	0	...
768	0	0	0	0	...
774	0	0	0	0	...
780	0	0	0	0	...
899	0	0	0	0	...
1015	0	0	0	0	...
1020	0	0	0	0	...
1063	0	0	0	0	...
...	...	...	...	...	...
84695	0	0	0	0	...
84698	0	0	0	0	...
84739	0	0	0	0	...
84827	0	0	0	0	...
84828	0	0	0	0	...
84843	0	0	0	0	...
84871	0	0	0	0	...
84952	0	0	0	0	...
85734	0	0	0	0	...
86820	0	0	0	0	...
86842	0	0	0	0	...
86845	0	0	0	0	...
86862	0	0	0	0	...
87041	0	0	0	0	...
87139	0	0	0	0	...
87297	0	0	0	0	...
87363	0	0	0	0	...
87557	0	0	0	0	...
87762	0	0	0	0	...
87911	0	0	0	0	...
88111	0	0	0	0	...
88240	0	0	0	0	...
88378	0	0	0	0	...
88571	0	0	0	0	...

88733	0	0	0	0	...
88797	0	0	0	0	...
89057	0	0	0	0	...
89060	0	0	0	0	...
89697	0	0	0	0	...
90006	0	0	0	0	...

	Area Control / Area Influence	Area Enclosure	Area Movement	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	0	0	0	
447	0	0	0	
504	0	0	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	
597	0	0	0	
611	0	0	0	
619	0	0	0	
629	0	0	0	
644	0	0	0	
650	0	0	0	
668	0	0	0	
768	0	0	0	
774	0	0	0	
780	0	0	0	
899	0	0	0	
1015	0	0	0	
1020	0	0	0	
1063	0	0	0	
...	...	...	...	
84695	0	0	0	
84698	0	0	0	
84739	0	0	0	
84827	0	0	0	
84828	0	0	0	
84843	0	0	1	
84871	1	0	0	
84952	0	0	0	
85734	0	0	0	

86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	1	0	1
87041	0	0	0
87139	0	0	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	1	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	0

	Area-Impulse	Auction/Bidding	Betting/Wagering	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	1	0	
402	0	0	0	
430	0	1	0	
447	0	0	0	
504	0	1	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	
597	0	0	0	
611	0	0	0	
619	0	0	0	
629	0	0	0	
644	0	0	0	
650	0	0	0	
668	0	0	0	
768	0	0	0	
774	0	0	0	

780	0	0	1
899	0	0	0
1015	0	0	0
1020	0	0	1
1063	0	0	0
...	...	...	...
84695	0	0	0
84698	0	0	0
84739	0	0	0
84827	0	0	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	0
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	1	0
87139	0	0	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	0

	Campaign / Battle Card Driven	Card Drafting	Chit-Pull System	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	0	0	0	
447	0	0	0	
504	0	0	0	

508	0	0	0
511	0	0	0
526	0	0	0
551	0	0	0
571	0	0	0
575	0	0	0
597	0	1	0
611	0	0	0
619	0	0	0
629	0	0	0
644	0	0	0
650	0	0	0
668	0	0	0
768	0	0	0
774	0	0	0
780	0	0	0
899	0	0	0
1015	0	0	0
1020	0	0	0
1063	0	0	0
...	...	...	...
84695	1	0	0
84698	0	0	0
84739	0	0	0
84827	0	1	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	0
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	0	0
87139	0	0	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	1	0
88733	0	0	0
88797	0	0	1
89057	0	0	0

89060	0	1	0
89697	1	0	0
90006	0	0	0

	Co-operative Play	Commodity Speculation	Crayon Rail System	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	0	0	0	
447	0	0	0	
504	0	0	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	
597	0	0	0	
611	0	0	0	
619	0	0	0	
629	0	0	0	
644	0	0	0	
650	0	0	0	
668	0	0	0	
768	0	0	0	
774	0	0	0	
780	0	0	0	
899	0	0	0	
1015	0	0	0	
1020	0	0	0	
1063	0	0	0	
...	...	...	...	
84695	0	0	0	
84698	1	0	0	
84739	0	0	0	
84827	0	0	0	
84828	0	0	0	
84843	1	0	0	
84871	0	0	0	
84952	0	0	0	
85734	0	0	0	
86820	0	0	0	
86842	1	0	0	
86845	0	0	0	

86862	0	0	0
87041	0	1	0
87139	0	0	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	0

	Deck / Pool Building	Dice Rolling	Grid Movement	Hand Management	\
100	0	0	0	0	
133	0	0	0	0	
152	0	1	0	0	
221	0	0	0	0	
258	0	0	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	0	0	0	
511	0	0	0	0	
526	0	0	0	0	
551	0	0	0	0	
571	0	0	0	0	
575	0	0	0	0	
597	0	1	0	0	
611	0	1	0	0	
619	0	0	0	0	
629	0	1	0	0	
644	0	0	0	0	
650	0	0	0	0	
668	0	0	0	0	
768	0	0	0	0	
774	0	0	0	0	
780	0	0	0	0	
899	0	0	0	0	
1015	0	0	0	0	

1020	0	0	0	0
1063	0	0	0	0
...	...	...	...	...
84695	0	1	0	0
84698	0	0	0	1
84739	1	1	1	0
84827	0	0	0	1
84828	0	1	0	0
84843	0	1	0	1
84871	0	0	0	0
84952	0	0	0	0
85734	0	1	0	0
86820	0	0	0	1
86842	1	0	0	1
86845	0	0	0	0
86862	0	0	0	0
87041	0	0	0	0
87139	0	0	0	0
87297	0	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	1
87911	0	0	0	0
88111	0	0	0	0
88240	0	1	0	0
88378	0	0	0	1
88571	0	0	0	0
88733	0	0	0	0
88797	0	0	0	0
89057	0	0	1	0
89060	1	0	0	1
89697	0	1	0	0
90006	0	1	1	0

	Hex-and-Counter	Line Drawing	Memory_mechanics	Modular Board	\
100	0	0	0	1	
133	0	0	0	0	
152	0	0	0	0	
221	0	0	0	0	
258	1	0	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	0	0	0	
511	0	0	0	1	
526	0	0	0	0	

551	0	0	0	0
571	0	0	0	0
575	0	0	0	0
597	0	0	0	0
611	0	0	0	0
619	0	0	0	0
629	1	0	0	0
644	0	0	0	0
650	1	0	0	0
668	0	0	0	0
768	0	0	0	0
774	0	0	0	0
780	0	0	0	0
899	1	0	0	0
1015	0	0	0	0
1020	0	0	0	0
1063	0	0	0	0
...	...	...	...	...
84695	0	0	0	0
84698	0	0	0	0
84739	0	0	0	0
84827	0	0	0	0
84828	0	0	0	0
84843	0	0	0	0
84871	0	0	0	0
84952	0	0	0	0
85734	0	0	0	1
86820	0	0	0	0
86842	0	0	0	0
86845	0	0	0	0
86862	0	0	0	0
87041	0	0	0	0
87139	1	0	0	0
87297	0	0	0	0
87363	0	0	1	0
87557	0	0	0	0
87762	0	0	0	0
87911	1	0	0	0
88111	0	0	0	0
88240	0	0	0	0
88378	0	0	0	0
88571	0	0	0	0
88733	0	0	0	0
88797	1	0	0	0
89057	0	0	0	1
89060	0	0	0	0
89697	1	0	0	0
90006	0	0	0	1

	Paper-and-Pencil	Partnerships	Pattern Building	Pattern Recognition	\
100	0	0	0	0	
133	0	0	0	0	
152	0	0	0	0	
221	0	0	0	0	
258	0	0	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	1	0	0	
511	0	0	0	0	
526	0	0	0	0	
551	0	0	1	0	
571	0	0	0	0	
575	0	0	0	0	
597	0	0	0	0	
611	0	0	0	0	
619	0	0	0	0	
629	0	0	0	0	
644	0	0	0	0	
650	0	0	0	0	
668	0	0	0	0	
768	0	0	0	0	
774	0	0	0	0	
780	0	0	0	0	
899	0	0	0	0	
1015	0	0	0	0	
1020	0	0	0	0	
1063	0	1	0	0	
...	...	...	...	...	
84695	0	0	0	0	
84698	0	0	0	0	
84739	0	0	0	0	
84827	0	0	0	0	
84828	0	0	0	0	
84843	0	0	0	0	
84871	0	0	0	0	
84952	0	0	0	0	
85734	0	0	0	0	
86820	0	0	0	0	
86842	0	0	0	0	
86845	0	0	0	0	
86862	0	0	0	0	
87041	0	0	0	0	
87139	0	0	0	0	

87297	0	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	0
87911	0	0	0	0
88111	0	0	0	0
88240	0	0	0	0
88378	0	0	0	0
88571	0	0	0	0
88733	0	0	1	0
88797	0	0	0	0
89057	0	0	0	0
89060	0	0	0	0
89697	0	0	0	0
90006	0	0	0	0

	Pick-up and Deliver	Player Elimination	Point to Point Movement	\
100	0	0	0	
133	0	0	0	
152	0	0	1	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	0	0	0	
447	0	0	0	
504	0	0	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	
597	0	0	0	
611	0	0	0	
619	0	0	0	
629	0	0	0	
644	0	0	0	
650	0	0	0	
668	0	0	0	
768	0	0	0	
774	0	0	0	
780	0	0	0	
899	0	0	0	
1015	0	0	0	
1020	0	0	0	
1063	0	0	0	
...	...	...	...	

84695	0	0	1
84698	0	0	0
84739	0	0	0
84827	0	0	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	0
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	0	0
87139	0	0	0
87297	0	0	0
87363	0	1	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	1
89060	0	0	0
89697	0	0	0
90006	0	0	0

	Press Your Luck	Rock-Paper-Scissors	Role Playing	\
100	0	0	0	
133	0	0	0	
152	0	0	0	
221	0	0	0	
258	0	0	0	
337	0	0	0	
402	0	0	0	
430	0	0	0	
447	0	0	0	
504	0	0	0	
508	0	0	0	
511	0	0	0	
526	0	0	0	
551	0	0	0	
571	0	0	0	
575	0	0	0	

597	0	0	0
611	0	0	0
619	0	0	0
629	0	0	0
644	0	0	0
650	0	0	0
668	0	0	0
768	0	0	0
774	0	0	0
780	0	0	0
899	0	0	0
1015	0	0	0
1020	0	0	0
1063	0	0	0
...	...	...	...
84695	0	0	0
84698	0	0	0
84739	0	0	0
84827	0	0	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	1
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	0	0
87139	0	0	0
87297	0	0	0
87363	0	0	0
87557	0	0	0
87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	1

	Roll / Spin and Move	Route/Network Building	Secret Unit Deployment	\
100	0	0	0	

133	0	0	0
152	0	0	0
221	0	0	0
258	0	0	0
337	0	0	0
402	0	0	0
430	0	0	0
447	0	0	0
504	0	0	0
508	0	0	0
511	0	0	0
526	0	0	0
551	0	0	0
571	0	0	0
575	0	0	0
597	0	0	0
611	0	0	0
619	0	0	0
629	0	0	0
644	0	0	0
650	0	0	0
668	0	0	0
768	0	0	0
774	0	0	0
780	0	0	0
899	0	0	0
1015	0	0	0
1020	0	0	0
1063	0	0	0
...	...	...	...
84695	0	0	0
84698	0	0	0
84739	0	0	0
84827	0	0	0
84828	0	0	0
84843	0	0	0
84871	0	0	0
84952	0	0	0
85734	0	0	0
86820	0	0	0
86842	0	0	0
86845	0	0	0
86862	0	0	0
87041	0	1	0
87139	0	0	0
87297	0	1	0
87363	0	0	0
87557	0	0	0

87762	0	0	0
87911	0	0	0
88111	0	0	0
88240	0	0	0
88378	0	0	0
88571	0	0	0
88733	0	0	0
88797	0	0	0
89057	0	0	0
89060	0	0	0
89697	0	0	0
90006	0	0	0

	Set Collection	Simulation	Simultaneous Action Selection	Singing	\
100	0	0	0	0	
133	0	0	0	0	
152	0	1	0	0	
221	0	0	0	0	
258	0	0	0	0	
337	0	0	0	0	
402	0	0	0	0	
430	0	0	0	0	
447	0	0	0	0	
504	0	0	0	0	
508	0	0	0	0	
511	0	0	0	0	
526	0	0	0	0	
551	0	0	0	0	
571	0	0	0	0	
575	0	0	0	0	
597	0	0	0	0	
611	0	0	0	0	
619	0	0	0	0	
629	0	0	0	0	
644	0	0	0	0	
650	0	0	0	0	
668	0	0	0	0	
768	0	0	0	0	
774	0	0	0	0	
780	0	0	0	0	
899	0	0	0	0	
1015	0	0	0	0	
1020	1	0	0	0	
1063	0	0	0	0	
...	...	...	...	...	
84695	0	0	0	0	
84698	0	0	0	0	
84739	0	0	0	0	

84827	1	0	0	0
84828	0	0	1	0
84843	0	0	0	0
84871	0	0	0	0
84952	0	0	0	0
85734	0	0	0	0
86820	1	0	0	0
86842	0	0	0	0
86845	0	0	0	0
86862	0	0	1	0
87041	0	0	0	0
87139	0	0	0	0
87297	1	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	0
87911	0	0	0	0
88111	0	0	0	0
88240	0	0	0	0
88378	0	0	0	0
88571	0	0	0	0
88733	0	0	0	0
88797	0	0	0	0
89057	0	0	0	0
89060	0	0	0	0
89697	0	1	0	0
90006	0	0	0	0

	Stock Holding	Storytelling	Take That	Tile Placement	Time Track	\
100	0	0	0	0	0	
133	0	0	1	0	0	
152	0	0	0	0	0	
221	0	0	0	0	0	
258	0	0	0	0	0	
337	0	0	0	0	0	
402	0	0	0	0	0	
430	0	0	0	0	0	
447	0	0	0	0	0	
504	0	0	0	0	0	
508	0	0	0	0	0	
511	0	0	0	0	0	
526	0	0	0	1	0	
551	0	0	0	1	0	
571	0	0	0	0	0	
575	0	0	0	0	0	
597	0	0	1	0	0	
611	0	0	0	0	0	
619	0	0	0	0	0	

629	0	0	0	0	0
644	0	0	0	0	0
650	0	0	0	0	0
668	0	0	0	0	0
768	0	0	0	0	0
774	0	0	0	0	0
780	0	0	0	0	0
899	0	0	0	0	0
1015	0	0	0	0	0
1020	0	0	0	0	0
1063	0	0	0	0	0
...	...	...	...	...	...
84695	0	0	0	0	0
84698	0	0	0	0	0
84739	0	0	1	0	0
84827	0	0	0	0	0
84828	0	0	0	0	0
84843	0	0	0	0	0
84871	0	0	0	0	0
84952	0	0	0	0	0
85734	0	0	0	0	0
86820	0	0	1	0	0
86842	0	0	0	0	0
86845	0	0	0	0	0
86862	0	0	0	0	0
87041	0	0	0	0	0
87139	0	0	0	0	0
87297	0	0	0	0	0
87363	0	0	1	0	0
87557	0	0	0	0	0
87762	0	0	0	0	0
87911	0	0	0	0	0
88111	0	0	0	0	0
88240	0	0	0	0	0
88378	0	0	0	0	0
88571	0	0	0	1	0
88733	0	0	0	0	0
88797	0	0	0	0	0
89057	0	0	0	0	0
89060	0	0	0	0	0
89697	0	0	0	0	0
90006	0	1	0	0	0

	Trading	Trick-taking	Variable Phase Order	Variable Player Powers	\
100	0	0	0	0	
133	0	0	0	0	
152	0	0	0	1	
221	0	0	0	0	

258	0	0	0	0
337	0	0	0	0
402	0	1	0	0
430	0	0	0	0
447	0	0	0	0
504	0	0	0	0
508	0	0	0	0
511	0	0	0	0
526	0	0	0	0
551	0	0	0	0
571	0	1	0	0
575	0	0	0	0
597	0	0	0	0
611	0	1	0	0
619	0	0	0	0
629	0	0	0	0
644	0	0	0	0
650	0	0	0	0
668	0	0	0	0
768	0	0	0	0
774	0	0	0	0
780	0	0	0	0
899	0	0	0	0
1015	0	0	0	0
1020	0	0	0	0
1063	0	0	0	0
...	...	...	...	...
84695	0	0	0	0
84698	0	0	0	1
84739	0	0	0	0
84827	0	0	0	0
84828	0	0	0	0
84843	0	0	0	0
84871	0	0	0	0
84952	0	0	0	0
85734	0	0	0	0
86820	0	0	0	0
86842	0	0	0	0
86845	0	0	0	0
86862	0	0	1	0
87041	0	0	0	0
87139	0	0	0	0
87297	0	0	0	0
87363	0	0	0	0
87557	0	0	0	0
87762	0	0	0	0
87911	0	0	0	0
88111	0	1	0	0

88240	0	0	0	1
88378	0	0	0	1
88571	0	0	0	0
88733	0	0	0	0
88797	0	0	0	0
89057	0	0	0	0
89060	0	0	0	0
89697	0	0	0	0
90006	0	0	0	1

	Voting	Worker Placement	binned.average	binary.success
100	0	0	5.0	0
133	0	0	4.0	0
152	0	0	5.0	0
221	0	0	4.0	0
258	0	0	7.0	1
337	0	0	6.0	0
402	0	0	6.0	0
430	0	0	5.0	0
447	0	0	5.0	0
504	0	0	6.0	0
508	0	0	5.0	0
511	0	0	4.0	0
526	0	0	6.0	0
551	0	0	5.0	0
571	0	0	6.0	0
575	0	0	6.0	0
597	0	0	6.0	0
611	0	0	7.0	1
619	0	0	6.0	0
629	0	0	6.0	0
644	0	0	5.0	0
650	0	0	6.0	0
668	0	0	5.0	0
768	0	0	6.0	0
774	0	0	4.0	0
780	0	0	6.0	0
899	0	0	6.0	0
1015	0	0	5.0	0
1020	0	0	4.0	0
1063	0	0	5.0	0
...	...	...	...	...
84695	0	0	8.0	1
84698	0	0	7.0	1
84739	0	0	7.0	1
84827	0	0	6.0	0
84828	0	0	6.0	0
84843	0	0	8.0	1

```
84871       0                 1              6.0                 0
84952       0                 0              7.0                 1
85734       0                 0              8.0                 1
86820       0                 0              7.0                 1
86842       0                 0              7.0                 1
86845       0                 0              6.0                 0
86862       0                 0              7.0                 1
87041       0                 0              7.0                 1
87139       0                 0              7.0                 1
87297       0                 0              7.0                 1
87363       0                 0              5.0                 0
87557       0                 0              6.0                 0
87762       0                 0              7.0                 1
87911       0                 0              8.0                 1
88111       0                 0              6.0                 0
88240       0                 0              7.0                 1
88378       0                 0              7.0                 1
88571       0                 0              7.0                 1
88733       0                 0              7.0                 1
88797       0                 0              8.0                 1
89057       0                 0              8.0                 1
89060       0                 0              7.0                 1
89697       0                 0              8.0                 1
90006       0                 0              8.0                 1

[2089 rows x 147 columns]
```

```
In [40]: df_data.loc[df_data['details.minage']==42]
```

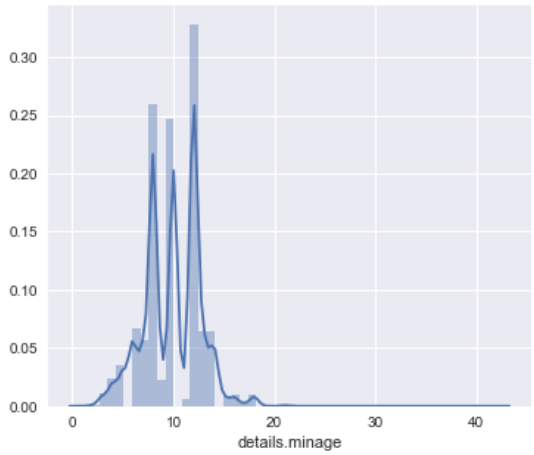
```
Out[40]:                  details.name  details.maxplayers  details.minage  \
50948  South African Railroads                6.0            42.0

       details.minplayers  details.playingtime  details.yearpublished  \
50948                 3.0                 40.0                 2011.0

       stats.averageweight  stats.average  stats.owned  stats.stddev  \
50948               3.1667        7.58116         89.0       1.24822

       Abstract Strategy  Action / Dexterity  Adventure  Age of Reason  \
50948                  0                   0          0              0

       American Civil War  American Indian Wars  American Revolutionary War  \
50948                   0                     0                           0

       American West  Ancient  Animals  Arabian  Aviation / Flight  Bluffing  \
50948              0        0        0        0                  0         0

       Book  Card Game  Children's Game  City Building  Civil War  \
```

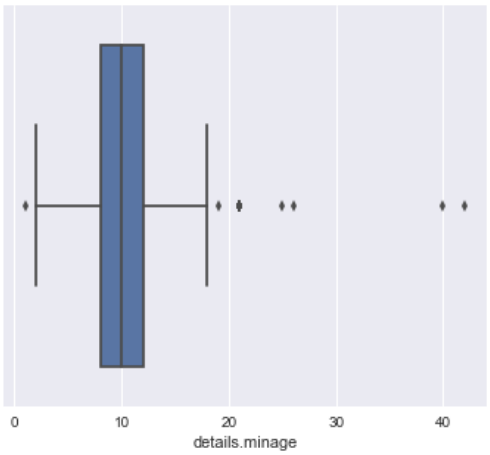
```
50948    0          0                0              0          0

       Civilization  Collectible Components  Comic Book / Strip  Deduction  \
50948             0                       0                   0          0

       Dice  Economic  Educational  Electronic  Environmental  \
50948     0         1            0           0              0

       Expansion for Base-game  Exploration  Fan Expansion  Fantasy  Farming  \
50948                        0            0              0        0        0

       Fighting  Game System  Horror  Humor  Industry / Manufacturing  \
50948         0            0       0      0                         0

       Korean War  Mafia  Math       ...        Area Control / Area Influence  \
50948           0      0     0       ...                                    0

       Area Enclosure  Area Movement  Area-Impulse  Auction/Bidding  \
50948               0              0             0                1

       Betting/Wagering  Campaign / Battle Card Driven  Card Drafting  \
50948                 0                              0              0

       Chit-Pull System  Co-operative Play  Commodity Speculation  \
50948                 0                  0                      0

       Crayon Rail System  Deck / Pool Building  Dice Rolling  Grid Movement  \
50948                   0                     0             0              0

       Hand Management  Hex-and-Counter  Line Drawing  Memory_mechanics  \
50948                0                 0             0                 0

       Modular Board  Paper-and-Pencil  Partnerships  Pattern Building  \
50948              0                 0             0                 0

       Pattern Recognition  Pick-up and Deliver  Player Elimination  \
50948                    0                    0                   0

       Point to Point Movement  Press Your Luck  Rock-Paper-Scissors  \
50948                        0                0                    0

       Role Playing  Roll / Spin and Move  Route/Network Building  \
50948             0                     0                       0

       Secret Unit Deployment  Set Collection  Simulation  \
50948                       0               0           0

       Simultaneous Action Selection  Singing  Stock Holding  Storytelling  \
```

```
50948                              0        0              1             0

       Take That  Tile Placement  Time Track  Trading  Trick-taking  \
50948          0               0           0        0             0

       Variable Phase Order  Variable Player Powers  Voting  Worker Placement  \
50948                     0                       0       0                 0

       binned.average  binary.success
50948             7.0               1

[1 rows x 147 columns]
```

We observe some interesting records, with a non-trival number of games that have a minimum age of 0, and one game, the infamous game, "South African Railroads," at the maximum extremes. As there are some board games that might be suitable for children, to distinguish between games that have been mis-inputted, we will set the minimum age to 1. While not thrilled about including South African Railroad's 42 year minimum age in (but very much morbidly curious), we will nevertheless keep it in.

```
In [41]: df_data2 = df_data1[(df_data1['details.minage']>=1)]

In [42]: df_data1['details.minage'].count()-df_data2['details.minage'].count()

Out[42]: 1971

In [43]: df_data2['details.minage'].count()

Out[43]: 18396

In [44]: analyze_feature(df_data2, "details.minage")

count    18396.000000
mean         9.952327
std          2.788389
min          1.000000
25%          8.000000
50%         10.000000
75%         12.000000
max         42.000000
Name: details.minage, dtype: float64
```



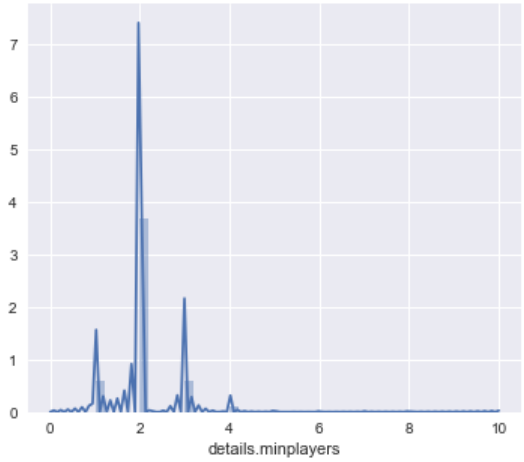


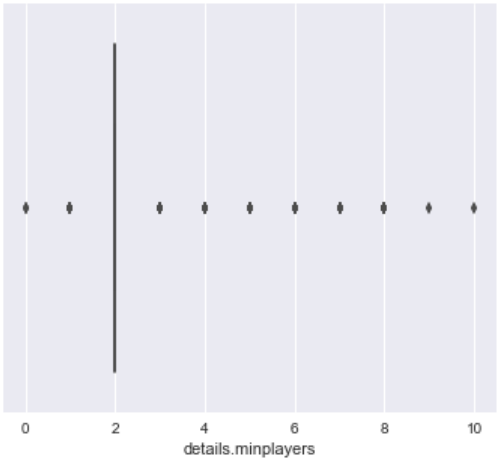
We eliminated 1971 games based upon the parameter that the value of minimum age was zero, and are now left with 18396 records.

MinPlayers

```
In [45]: analyze_feature(df_data2, "details.minplayers")
```

```
count    18396.000000
mean         2.070450
std          0.662213
min          0.000000
25%          2.000000
50%          2.000000
75%          2.000000
max         10.000000
Name: details.minplayers, dtype: float64
```





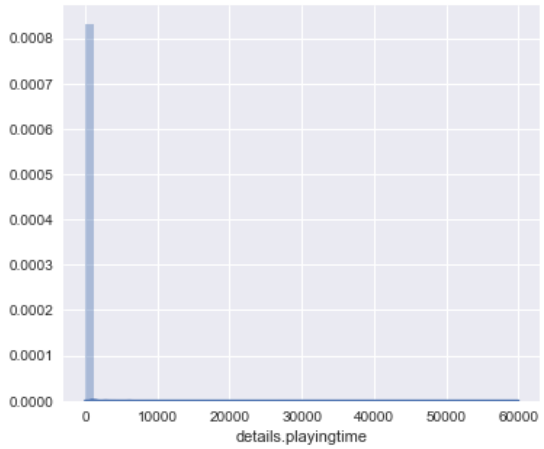
We still see the issue that somes games have a minimum number of players that are equivalent to zero. As stated above, we believe that this does not make logical sense and we will filter these games out.

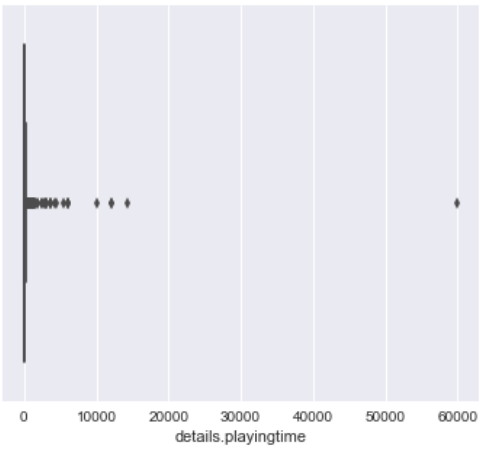
```
In [46]: df_data3 = df_data2[(df_data2['details.minplayers']>=1)]

In [47]: df_data2['details.minage'].count()-df_data3['details.minage'].count()

Out[47]: 4

In [48]: df_data3['details.minplayers'].describe()

Out[48]: count    18392.000000
         mean         2.070900
         std          0.661581
         min          1.000000
         25%          2.000000
         50%          2.000000
         75%          2.000000
         max         10.000000
         Name: details.minplayers, dtype: float64
```

We have filtered out four games and now have 18392 records

Playing Time

```
In [49]: analyze_feature(df_data3, "details.playingtime")

count    18392.000000
mean        75.952371
std        499.378068
min          0.000000
25%         20.000000
50%         45.000000
75%         90.000000
max      60000.000000
Name: details.playingtime, dtype: float64
```






```
In [50]: df_data3.loc[df_data['details.playingtime']==60000]
```

Out[50]:

	details.name	details.maxplayers	details.minage	details.minplayers	details.playingtime	details.yearpublished	stats.averageweight	stats.average	stats.owned	stats.stddev	Abstract Strategy	Action / Dexterity	Adventure	Age of Reason	American Civil War	American Indian Wars	American Revolutionary War	American West	Ancient	Animals	Arabian	Aviation / Flight	Bluffing	Book	Card Game	Children's Game	City Building	Civil War	Civilization	Collectible Components	Comic Book / Strip	Deduction	Dice	Economic	Educational	Electronic	Environmental	Expansion for Base-game	Exploration	Fan Expansion	Fantasy	Farming
4360	The Campaign for North Africa	10.0	14.0	8.0	60000.0	1979.0	5.0	5.62887	283.0	2.77789	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

```
      Fighting  Game System  Horror  Humor  Industry / Manufacturing  \
4360         0            0       0      0                         0

      Korean War  Mafia  Math       ...        Area Control / Area Influence  \
4360           0      0     0       ...                                     0

      Area Enclosure  Area Movement  Area-Impulse  Auction/Bidding  \
4360               0              0             0                0

      Betting/Wagering  Campaign / Battle Card Driven  Card Drafting  \
4360                 0                              0              0

      Chit-Pull System  Co-operative Play  Commodity Speculation  \
4360                 0                  0                      0

      Crayon Rail System  Deck / Pool Building  Dice Rolling  Grid Movement  \
4360                   0                     0             0              0

      Hand Management  Hex-and-Counter  Line Drawing  Memory_mechanics  \
4360                0                1             0                 0

      Modular Board  Paper-and-Pencil  Partnerships  Pattern Building  \
4360              0                 0             0                 0

      Pattern Recognition  Pick-up and Deliver  Player Elimination  \
4360                   0                    0                   0

      Point to Point Movement  Press Your Luck  Rock-Paper-Scissors  \
4360                       0                0                    0

      Role Playing  Roll / Spin and Move  Route/Network Building  \
4360             0                     0                       0

      Secret Unit Deployment  Set Collection  Simulation  \
4360                       0               0           0

      Simultaneous Action Selection  Singing  Stock Holding  Storytelling  \
4360                              0        0              0             0

      Take That  Tile Placement  Time Track  Trading  Trick-taking  \
4360         0               0           0        0             0

      Variable Phase Order  Variable Player Powers  Voting  Worker Placement  \
4360                     0                       0       0                 0

      binned.average  binary.success
4360             5.0               0
```

```
[1 rows x 147 columns]
```

```
In [51]: df_data3.loc[df_data3['details.playingtime']==0]
```

Out[51]:

	details.name	details.maxplayers	\
20	Gateway to the Stars	7.0	
50	Brauerei	6.0	
52	Sophie's World	6.0	
250	Neue Spiele im alten Rom	7.0	
251	Ostindiska Kompaniet	5.0	
439	Pacal	2.0	
482	Ice Blocks	2.0	
489	Flohzirkus	4.0	
491	Aerodrome	12.0	
581	Vox Populi	6.0	
664	War of Resistance	2.0	
676	Lunatix Loop	6.0	
681	Formula C Minus	6.0	
724	Amoeba	4.0	
771	Fire	5.0	
819	Corner	4.0	
864	1848	2.0	
871	Panda Monium	6.0	
1004	Dow Jones	7.0	
1163	Legend of Heroes	5.0	
1191	Abenteuer im Wichtelwald	4.0	
1192	Aber Hallo!	4.0	
1195	Alles klar?	5.0	
1200	Basta!	6.0	
1775	Votum	6.0	
1783	The Gothic Game	8.0	
1787	Steinbeisser	6.0	
2216	Chaos Tiles	5.0	
2217	The Chicago Way	4.0	
2226	Express Chess	2.0	
...	...	...	
87579	Avec Infini Regret II	2.0	
87655	Lady Richmond: Ein erzocktes Erbe	5.0	
87657	Drachenturm	4.0	
87679	Picassimo	6.0	
87681	Hamsterbande	4.0	
87714	Totem	4.0	
87963	SLAPZI	10.0	
87985	Doctor Who: Time Clash Starter Set	4.0	
88025	Magic: The Gathering Duel Decks: Elspeth vs...	2.0	
88028	Magic: The Gathering Duel Decks: Venser vs. ...	2.0	
88030	Magic: The Gathering Duel Decks: Sorin vs. T...	2.0	
88031	Magic: The Gathering Duel Decks: Heroes vs. ...	2.0	

```
88033  Magic: The Gathering  Duel Decks: Elspeth vs...                2.0
88034  Magic: The Gathering  Duel Decks: Nissa vs. O...                2.0
88209                                            Take That              4.0
88328                                         Balloon Pop!             10.0
88500                              Level 8: Das Kartenspiel              6.0
88505                                         Tzulan Quest              5.0
88624                                           Braintopia              6.0
88629                             The Legend of the Wendigo             6.0
88630                                              Pyramids             5.0
88698               The Terminator: The Official Board Game             5.0
88699             Ascension: Year Three Collector's Edition             6.0
88920                                      Watch Ya' Mouth             10.0
89080                                                IKAN              4.0
89546                               Qwinto: Das Kartenspiel             4.0
89549                                            Twenty One             6.0
89786                                  Deckscape: Test Time             6.0
89788                                              3 Secrets             8.0
89821                               Aeon's End: War Eternal              4.0

       details.minage  details.minplayers  details.playingtime  \
20               12.0                 1.0                  0.0
50               14.0                 3.0                  0.0
52               12.0                 2.0                  0.0
250              10.0                 2.0                  0.0
251              10.0                 2.0                  0.0
439              12.0                 2.0                  0.0
482              10.0                 2.0                  0.0
489               5.0                 2.0                  0.0
491              12.0                 2.0                  0.0
581              12.0                 3.0                  0.0
664              12.0                 2.0                  0.0
676              10.0                 4.0                  0.0
681              10.0                 2.0                  0.0
724               8.0                 2.0                  0.0
771               6.0                 2.0                  0.0
819              10.0                 2.0                  0.0
864              12.0                 2.0                  0.0
871               6.0                 3.0                  0.0
1004             12.0                 3.0                  0.0
1163              8.0                 1.0                  0.0
1191              8.0                 2.0                  0.0
1192             10.0                 3.0                  0.0
1195             10.0                 2.0                  0.0
1200              9.0                 2.0                  0.0
1775             14.0                 4.0                  0.0
1783             10.0                 2.0                  0.0
1787              7.0                 2.0                  0.0
2216              8.0                 1.0                  0.0
```

2217	14.0	3.0	0.0
2226	8.0	2.0	0.0
...	...	...	...
87579	12.0	2.0	0.0
87655	8.0	2.0	0.0
87657	5.0	2.0	0.0
87679	8.0	3.0	0.0
87681	4.0	1.0	0.0
87714	7.0	2.0	0.0
87963	8.0	2.0	0.0
87985	14.0	2.0	0.0
88025	13.0	2.0	0.0
88028	13.0	2.0	0.0
88030	13.0	2.0	0.0
88031	13.0	2.0	0.0
88033	13.0	2.0	0.0
88034	13.0	2.0	0.0
88209	8.0	2.0	0.0
88328	13.0	1.0	0.0
88500	10.0	2.0	0.0
88505	8.0	2.0	0.0
88624	8.0	2.0	0.0
88629	6.0	2.0	0.0
88630	10.0	2.0	0.0
88698	14.0	2.0	0.0
88699	13.0	1.0	0.0
88920	8.0	3.0	0.0
89080	8.0	2.0	0.0
89546	8.0	1.0	0.0
89549	8.0	2.0	0.0
89786	12.0	1.0	0.0
89788	14.0	2.0	0.0
89821	14.0	1.0	0.0

	details.yearpublished	stats.averageweight	stats.average	stats.owned	\
20	1981.0	3.0000	5.35714	80.0	
50	1996.0	3.0000	5.83148	85.0	
52	1998.0	1.5714	4.93768	184.0	
250	1994.0	2.1818	6.81040	390.0	
251	1991.0	2.5455	4.75000	196.0	
439	1999.0	1.6667	6.19779	132.0	
482	2000.0	1.5000	5.38462	21.0	
489	1997.0	1.0000	6.68182	28.0	
491	1994.0	2.1667	7.63393	60.0	
581	1999.0	2.1111	4.45733	200.0	
664	1998.0	4.5000	6.86226	230.0	
676	2000.0	2.0000	6.26538	69.0	
681	2000.0	1.0000	4.81034	75.0	

724	1975.0	2.0000	5.99359	88.0
771	1996.0	1.1429	6.09643	127.0
819	1980.0	1.6000	6.42632	95.0
864	1998.0	3.0000	4.54348	80.0
871	1994.0	1.0000	5.38559	229.0
1004	1993.0	0.0000	2.78571	24.0
1163	1987.0	1.4000	5.64634	129.0
1191	2000.0	0.0000	5.78947	46.0
1192	2000.0	1.8000	5.62963	59.0
1195	1999.0	2.0000	5.28947	39.0
1200	1994.0	1.0000	4.36667	29.0
1775	1987.0	2.0000	2.84615	56.0
1783	1992.0	1.3846	6.44103	99.0
1787	1999.0	0.0000	5.50000	22.0
2216	1999.0	2.0000	5.24146	101.0
2217	2000.0	1.5000	3.27308	74.0
2226	1996.0	0.0000	4.58333	85.0
...	...	...	...	...
87579	2016.0	3.0000	7.41176	77.0
87655	2016.0	2.0000	5.75000	50.0
87657	2016.0	1.3333	6.39259	44.0
87679	2016.0	2.0000	6.32143	49.0
87681	2016.0	1.0000	5.80769	34.0
87714	2016.0	0.0000	4.80263	15.0
87963	2015.0	1.0000	6.16538	31.0
87985	2016.0	2.6667	6.02500	63.0
88025	2010.0	0.0000	7.52727	23.0
88028	2012.0	0.0000	7.34545	29.0
88030	2013.0	0.0000	8.27273	36.0
88031	2013.0	0.0000	7.63846	54.0
88033	2015.0	4.0000	8.20833	51.0
88034	2016.0	0.0000	7.59615	68.0
88209	2016.0	2.0000	6.34615	108.0
88328	2017.0	1.0000	6.37143	85.0
88500	2016.0	1.0000	5.00000	98.0
88505	2016.0	2.0000	5.81818	28.0
88624	2017.0	1.5000	7.12702	121.0
88629	2017.0	0.0000	6.64706	43.0
88630	2017.0	2.0000	7.02499	221.0
88698	2017.0	0.0000	8.43478	30.0
88699	2016.0	0.0000	8.81579	99.0
88920	2016.0	0.0000	5.15618	95.0
89080	2017.0	2.0000	9.36667	0.0
89546	2017.0	0.0000	6.74375	31.0
89549	2017.0	1.0000	6.37356	126.0
89786	2017.0	1.0000	7.32069	70.0
89788	2017.0	2.0000	6.46364	38.0
89821	2017.0	0.0000	9.28333	56.0

	stats.stddev	Abstract Strategy	Action / Dexterity	Adventure	\
20	1.673620	0	0	0	
50	1.339370	0	0	0	
52	1.958640	0	0	0	
250	1.359410	1	0	0	
251	1.599580	0	0	0	
439	1.043010	0	0	0	
482	1.227160	1	0	0	
489	1.108900	0	1	0	
491	1.344330	0	0	0	
581	1.629740	0	0	0	
664	1.826870	0	0	0	
676	1.431590	0	0	0	
681	1.821390	0	0	0	
724	1.256380	1	0	0	
771	1.288200	0	1	0	
819	1.558790	1	0	0	
864	2.058460	0	0	0	
871	1.272990	0	1	0	
1004	1.566550	0	0	0	
1163	1.440880	0	0	1	
1191	1.150700	0	0	0	
1192	1.848890	0	0	0	
1195	1.228130	0	0	0	
1200	1.793200	0	0	0	
1775	1.511300	0	0	0	
1783	1.920170	0	0	0	
1787	1.066000	0	0	0	
2216	1.805270	1	0	0	
2217	1.588070	0	0	0	
2226	1.879860	1	0	0	
...	...	...	...	...	
87579	0.911290	0	0	0	
87655	1.462680	0	0	0	
87657	1.577840	0	1	0	
87679	1.304570	0	0	0	
87681	2.135220	0	0	0	
87714	1.273670	0	0	0	
87963	1.248290	0	1	0	
87985	1.945990	0	0	0	
88025	2.423030	0	0	0	
88028	2.528520	0	0	0	
88030	1.542780	0	0	0	
88031	1.713990	0	0	0	
88033	1.406210	0	0	0	
88034	2.344260	0	0	0	
88209	1.297850	0	0	0	

88328	1.104170	0	0	0
88500	1.710260	0	0	0
88505	2.308800	0	0	1
88624	0.977128	0	0	0
88629	1.233890	0	0	0
88630	1.152370	0	0	0
88698	2.410590	0	0	0
88699	0.846207	0	0	0
88920	1.742080	0	0	0
89080	0.865384	0	0	1
89546	1.669190	0	0	0
89549	1.381530	0	0	0
89786	0.882934	0	0	0
89788	1.270490	0	0	0
89821	1.737020	0	0	0

	Age of Reason	American Civil War	American Indian Wars	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	0	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	
864	0	0	0	
871	0	0	0	
1004	0	0	0	
1163	0	0	0	
1191	0	0	0	
1192	0	0	0	
1195	0	0	0	
1200	0	0	0	
1775	0	0	0	
1783	0	0	0	
1787	0	0	0	
2216	0	0	0	
2217	0	0	0	
2226	0	0	0	
...	...	...	...	

87579	0	0	0
87655	0	0	0
87657	0	0	0
87679	0	0	0
87681	0	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	0	0	0
88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	American Revolutionary War	American West	Ancient	Animals	Arabian	\
20	0	0	0	0	0	
50	0	0	0	0	0	
52	0	0	0	0	0	
250	0	0	0	0	0	
251	0	0	0	0	0	
439	0	0	0	0	0	
482	0	0	0	0	0	
489	0	0	0	0	0	
491	0	0	0	0	0	
581	0	0	0	0	0	
664	0	0	0	0	0	
676	0	0	0	0	0	
681	0	0	0	0	0	
724	0	0	0	0	0	
771	0	0	0	0	0	
819	0	0	0	0	0	

864	0	0	0	0	0
871	0	0	0	0	0
1004	0	0	0	0	0
1163	0	0	0	0	0
1191	0	0	0	0	0
1192	0	0	0	0	0
1195	0	0	0	0	0
1200	0	0	0	0	0
1775	0	0	0	0	0
1783	0	0	0	0	0
1787	0	0	0	0	0
2216	0	0	0	0	0
2217	0	0	0	0	0
2226	0	0	0	0	0
...	...	...	...	...	...
87579	0	0	0	0	0
87655	0	0	0	0	0
87657	0	0	0	0	0
87679	0	0	0	0	0
87681	0	0	0	1	0
87714	0	0	0	0	0
87963	0	0	0	0	0
87985	0	0	0	0	0
88025	0	0	0	0	0
88028	0	0	0	0	0
88030	0	0	0	0	0
88031	0	0	0	0	0
88033	0	0	0	0	0
88034	0	0	0	0	0
88209	0	0	0	0	0
88328	0	0	0	0	0
88500	0	0	0	0	0
88505	0	0	0	0	0
88624	0	0	0	0	0
88629	0	0	0	0	0
88630	0	0	0	0	0
88698	0	0	0	0	0
88699	0	0	0	0	0
88920	0	0	0	0	0
89080	0	0	0	0	0
89546	0	0	0	0	0
89549	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	0	0

	Aviation / Flight	Bluffing	Book	Card Game	Children's Game	\
20	0	0	0	0	0	

50	0	0	0	0	0
52	0	0	0	0	0
250	0	0	0	0	0
251	0	0	0	0	0
439	0	0	0	1	0
482	0	0	0	0	0
489	0	0	0	0	0
491	1	0	0	0	0
581	0	0	0	0	0
664	0	0	0	0	0
676	0	0	0	0	0
681	0	0	0	0	0
724	0	0	0	0	0
771	0	0	0	0	0
819	0	0	0	0	0
864	0	0	0	1	0
871	0	0	0	1	1
1004	0	0	0	0	0
1163	0	0	0	0	0
1191	0	0	0	0	0
1192	0	0	0	1	0
1195	0	0	0	1	0
1200	0	0	0	1	0
1775	0	0	0	0	0
1783	0	0	0	0	0
1787	0	0	0	0	0
2216	0	0	0	0	0
2217	0	0	0	0	0
2226	0	0	0	1	0
...	...	...	...	...	...
87579	0	0	0	0	0
87655	0	1	0	1	0
87657	0	0	0	0	1
87679	0	0	0	0	0
87681	0	0	0	0	1
87714	0	0	0	0	0
87963	0	0	0	1	1
87985	0	0	0	1	0
88025	0	0	0	1	0
88028	0	0	0	1	0
88030	0	0	0	1	0
88031	0	0	0	1	0
88033	0	0	0	1	0
88034	0	0	0	1	0
88209	0	0	0	1	0
88328	0	0	0	0	0
88500	0	0	0	1	0
88505	0	0	0	1	0

88624	0	0	0	0	0
88629	0	0	0	0	1
88630	0	0	0	1	0
88698	0	0	0	0	0
88699	0	0	0	1	0
88920	0	0	0	0	0
89080	0	0	0	0	0
89546	0	0	0	1	0
89549	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	1	0

	City Building	Civil War	Civilization	Collectible Components	\
20	0	0	1	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	
251	0	0	0	0	
439	0	0	0	0	
482	0	0	0	0	
489	0	0	0	0	
491	0	0	0	0	
581	0	0	0	0	
664	0	0	0	0	
676	0	0	0	0	
681	0	0	0	0	
724	0	0	0	0	
771	0	0	0	0	
819	0	0	0	0	
864	0	0	0	0	
871	0	0	0	0	
1004	0	0	0	0	
1163	0	0	0	0	
1191	0	0	0	0	
1192	0	0	0	0	
1195	0	0	0	0	
1200	0	0	0	0	
1775	0	0	0	0	
1783	0	0	0	0	
1787	0	0	0	0	
2216	0	0	0	0	
2217	0	0	0	0	
2226	0	0	0	0	
...	...	...	...	...	
87579	0	0	0	0	
87655	0	0	0	0	
87657	0	0	0	0	

87679	0	0	0	0
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	0	0	0	0
88505	0	0	0	0
88624	0	0	0	0
88629	0	0	0	0
88630	0	0	0	0
88698	0	0	0	0
88699	0	0	0	0
88920	0	0	0	0
89080	0	0	0	0
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0

	Comic Book / Strip	Deduction	Dice	Economic	Educational	Electronic	\
20	0	0	0	0	0	0	
50	0	0	0	1	0	0	
52	0	0	0	0	0	0	
250	0	0	0	0	0	0	
251	0	0	0	1	0	0	
439	0	0	0	0	0	0	
482	0	0	0	0	0	0	
489	0	0	0	0	0	0	
491	0	0	0	0	0	0	
581	0	0	0	0	0	0	
664	0	0	0	0	0	0	
676	0	0	0	0	0	0	
681	0	0	0	0	0	0	
724	0	0	0	0	0	0	
771	0	0	0	0	0	0	
819	0	0	0	0	0	0	
864	0	0	0	0	0	0	
871	0	0	0	0	0	0	
1004	0	0	0	0	0	0	

1163	0	0	0	0	0	0
1191	0	0	0	0	0	0
1192	0	0	0	0	0	0
1195	0	0	0	0	0	0
1200	0	0	0	0	0	0
1775	0	0	0	0	0	0
1783	0	0	0	0	0	0
1787	0	0	0	0	0	0
2216	0	0	0	0	0	0
2217	0	0	0	0	0	0
2226	0	0	0	0	0	0
...	...	...	...	...	...	...
87579	0	0	0	0	0	0
87655	0	0	0	0	0	0
87657	0	0	0	0	0	0
87679	0	1	0	0	0	0
87681	0	0	1	0	0	0
87714	0	0	0	0	0	0
87963	0	0	0	0	1	0
87985	0	0	0	0	0	0
88025	0	0	0	0	0	0
88028	0	0	0	0	0	0
88030	0	0	0	0	0	0
88031	0	0	0	0	0	0
88033	0	0	0	0	0	0
88034	0	0	0	0	0	0
88209	0	0	0	0	0	0
88328	0	0	1	0	0	0
88500	0	0	0	0	0	0
88505	0	0	0	0	0	0
88624	0	0	0	0	0	0
88629	0	0	0	0	0	0
88630	0	0	0	0	0	0
88698	0	0	0	0	0	0
88699	0	0	0	0	0	0
88920	0	0	0	0	0	0
89080	0	0	0	0	0	0
89546	0	0	0	0	0	0
89549	0	0	1	0	0	0
89786	0	0	0	0	0	0
89788	0	1	0	0	0	0
89821	0	0	0	0	0	0

	Environmental	Expansion for Base-game	Exploration	Fan Expansion	\
20	0	0	1	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	

251	0	0	0	0
439	0	0	0	0
482	0	0	0	0
489	0	0	0	0
491	0	0	0	0
581	0	0	0	0
664	0	0	0	0
676	0	0	0	0
681	0	0	0	0
724	0	0	0	0
771	0	0	0	0
819	0	0	0	0
864	0	0	0	0
871	0	0	0	0
1004	0	0	0	0
1163	0	0	0	0
1191	0	0	0	0
1192	0	0	0	0
1195	0	0	0	0
1200	0	0	0	0
1775	0	0	0	0
1783	0	0	0	0
1787	0	0	0	0
2216	0	0	0	0
2217	0	0	0	0
2226	0	0	0	0
...	...	...	...	...
87579	0	0	0	0
87655	0	0	0	0
87657	0	0	0	0
87679	0	0	0	0
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	0	0	0	0
88505	0	0	1	0
88624	0	0	0	0
88629	0	0	0	0
88630	0	0	0	0

88698	0	0	0	0
88699	0	0	0	0
88920	0	0	0	0
89080	0	0	0	0
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0

	Fantasy	Farming	Fighting	Game System	Horror	Humor	\
20	0	0	0	0	0	0	
50	0	0	0	0	0	0	
52	0	0	0	0	0	0	
250	0	0	0	1	0	0	
251	0	0	0	0	0	0	
439	0	0	0	0	0	0	
482	0	0	0	0	0	0	
489	0	0	0	0	0	0	
491	0	0	0	0	0	0	
581	0	0	1	0	0	0	
664	0	0	0	0	0	0	
676	0	0	0	0	0	0	
681	0	0	0	0	0	1	
724	0	0	0	0	0	0	
771	0	0	0	0	0	0	
819	0	0	0	0	0	0	
864	0	0	0	0	0	0	
871	0	0	0	0	0	0	
1004	0	0	0	0	0	0	
1163	1	0	0	0	0	0	
1191	0	0	0	0	0	0	
1192	0	0	0	0	0	0	
1195	0	0	0	0	0	0	
1200	0	0	0	0	0	0	
1775	0	0	0	0	0	0	
1783	0	0	0	0	1	1	
1787	0	0	0	0	0	0	
2216	0	0	0	0	0	0	
2217	0	0	0	0	0	0	
2226	0	0	0	0	0	0	
...	...	...	...	...	...	...	
87579	0	0	0	0	0	0	
87655	0	0	0	0	0	1	
87657	1	0	0	0	0	0	
87679	0	0	0	0	0	0	
87681	0	0	0	0	0	0	
87714	0	0	0	0	0	0	

87963	0	0	0	0	0	0
87985	0	0	0	0	0	0
88025	1	0	1	0	0	0
88028	1	0	1	0	0	0
88030	1	0	1	0	0	0
88031	1	0	1	0	0	0
88033	1	0	1	0	0	0
88034	1	0	1	0	0	0
88209	0	0	0	0	0	0
88328	0	0	0	0	0	0
88500	0	0	0	0	0	0
88505	0	0	0	0	0	0
88624	0	0	0	0	0	0
88629	0	0	0	0	0	0
88630	0	0	0	0	0	0
88698	0	0	0	0	0	0
88699	1	0	0	0	0	0
88920	0	0	0	0	0	0
89080	0	0	0	0	0	0
89546	0	0	0	0	0	0
89549	0	0	0	0	0	0
89786	0	0	0	0	0	0
89788	0	0	0	0	0	0
89821	1	0	0	0	0	0

	Industry / Manufacturing	Korean War	Mafia	Math	...	\
20	0	0	0	0	...	
50	0	0	0	0	...	
52	0	0	0	0	...	
250	0	0	0	0	...	
251	0	0	0	0	...	
439	0	0	0	0	...	
482	0	0	0	0	...	
489	0	0	0	0	...	
491	0	0	0	0	...	
581	0	0	0	0	...	
664	0	0	0	0	...	
676	0	0	0	0	...	
681	0	0	0	0	...	
724	0	0	0	0	...	
771	0	0	0	0	...	
819	0	0	0	0	...	
864	0	0	0	0	...	
871	0	0	0	0	...	
1004	0	0	0	0	...	
1163	0	0	0	0	...	
1191	0	0	0	0	...	
1192	0	0	0	0	...	

1195	0	0	0	0	...
1200	0	0	0	0	...
1775	0	0	0	0	...
1783	0	0	0	0	...
1787	0	0	0	0	...
2216	0	0	0	0	...
2217	0	0	0	0	...
2226	0	0	0	0	...
...	...	...	...	...	...
87579	0	0	0	0	...
87655	0	0	0	0	...
87657	0	0	0	0	...
87679	0	0	0	0	...
87681	0	0	0	0	...
87714	0	0	0	0	...
87963	0	0	0	0	...
87985	0	0	0	0	...
88025	0	0	0	0	...
88028	0	0	0	0	...
88030	0	0	0	0	...
88031	0	0	0	0	...
88033	0	0	0	0	...
88034	0	0	0	0	...
88209	0	0	0	0	...
88328	0	0	0	0	...
88500	0	0	0	0	...
88505	0	0	0	0	...
88624	0	0	0	0	...
88629	0	0	0	0	...
88630	0	0	0	0	...
88698	0	0	0	0	...
88699	0	0	0	0	...
88920	0	0	0	0	...
89080	0	0	0	0	...
89546	0	0	0	0	...
89549	0	0	0	0	...
89786	0	0	0	0	...
89788	0	0	0	0	...
89821	0	0	0	0	...

	Area Control / Area Influence	Area Enclosure	Area Movement	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	

489	0	0	0
491	0	0	0
581	0	0	0
664	0	0	0
676	0	0	0
681	0	0	0
724	0	0	0
771	0	0	0
819	0	0	0
864	0	0	0
871	0	0	0
1004	0	0	0
1163	0	0	0
1191	0	0	0
1192	0	0	0
1195	0	0	0
1200	0	0	0
1775	0	0	0
1783	0	0	0
1787	0	0	0
2216	0	0	0
2217	0	0	0
2226	0	0	0
...	...	...	...
87579	0	0	0
87655	0	0	0
87657	0	0	0
87679	0	0	0
87681	0	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	0	0	0
88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	1
88699	0	0	0
88920	0	0	0

89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	Area-Impulse	Auction/Bidding	Betting/Wagering	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	0	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	
864	0	0	0	
871	0	0	0	
1004	0	0	0	
1163	0	0	0	
1191	0	0	0	
1192	0	0	0	
1195	0	0	0	
1200	0	0	0	
1775	0	0	0	
1783	0	0	0	
1787	0	0	0	
2216	0	0	0	
2217	0	0	0	
2226	0	0	0	
...	...	...	...	
87579	0	0	0	
87655	0	1	0	
87657	0	0	0	
87679	0	0	0	
87681	0	0	0	
87714	0	0	0	
87963	0	0	0	
87985	0	0	0	
88025	0	0	0	

88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	Campaign / Battle Card Driven	Card Drafting	Chit-Pull System	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	0	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	
864	0	0	0	
871	0	0	0	
1004	0	0	0	
1163	0	0	0	
1191	0	0	0	
1192	0	0	0	
1195	0	0	0	
1200	0	0	0	
1775	0	0	0	

1783	0	0	0
1787	0	0	0
2216	0	0	0
2217	0	0	0
2226	0	0	0
...	...	...	...
87579	0	0	0
87655	0	0	0
87657	0	0	0
87679	0	0	0
87681	0	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	1	0	0
88028	1	0	0
88030	1	0	0
88031	1	0	0
88033	1	0	0
88034	1	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	1	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	1	0

	Co-operative Play	Commodity Speculation	Crayon Rail System	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	1	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	

664	0	0	0
676	0	0	0
681	0	0	0
724	0	0	0
771	0	0	0
819	0	0	0
864	0	0	0
871	0	0	0
1004	0	0	0
1163	0	0	0
1191	0	0	0
1192	0	0	0
1195	0	0	0
1200	0	0	0
1775	0	0	0
1783	0	0	0
1787	0	0	0
2216	0	0	0
2217	0	0	0
2226	0	0	0
...	...	...	...
87579	0	0	0
87655	0	0	0
87657	1	0	0
87679	0	0	0
87681	1	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	0	0	0
88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	1	0	0
88699	0	0	0
88920	0	0	0
89080	1	0	0
89546	0	0	0
89549	0	0	0

89786	1	0	0
89788	1	0	0
89821	1	0	0

	Deck / Pool Building	Dice Rolling	Grid Movement	Hand Management	\
20	0	0	0	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	
251	0	0	0	0	
439	0	0	0	1	
482	0	0	0	0	
489	0	0	0	0	
491	0	0	0	0	
581	0	0	0	0	
664	0	0	0	0	
676	0	0	0	0	
681	0	1	0	1	
724	0	0	0	0	
771	0	0	0	0	
819	0	0	0	0	
864	0	0	0	0	
871	0	0	0	0	
1004	0	1	0	0	
1163	0	0	0	0	
1191	0	0	0	0	
1192	0	0	0	0	
1195	0	0	0	1	
1200	0	0	0	0	
1775	0	0	0	0	
1783	0	0	0	0	
1787	0	0	0	0	
2216	0	0	0	0	
2217	0	0	0	0	
2226	0	0	1	0	
...	...	...	...	...	
87579	0	1	0	0	
87655	0	0	0	0	
87657	0	0	0	0	
87679	0	0	0	0	
87681	0	0	0	0	
87714	0	0	0	0	
87963	0	0	0	0	
87985	0	0	0	0	
88025	0	0	0	1	
88028	0	0	0	1	
88030	0	0	0	1	
88031	0	0	0	1	

88033	0	0	0	1
88034	0	0	0	1
88209	0	0	0	1
88328	0	0	0	0
88500	0	0	0	0
88505	0	0	0	1
88624	0	0	0	0
88629	0	0	0	0
88630	0	0	0	0
88698	0	0	0	0
88699	1	0	0	1
88920	0	0	0	0
89080	0	0	0	0
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	1	0	0	1

	Hex-and-Counter	Line Drawing	Memory_mechanics	Modular Board	\
20	0	0	0	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	
251	0	0	0	0	
439	0	0	0	0	
482	0	0	0	0	
489	0	0	0	0	
491	0	0	0	0	
581	0	0	0	0	
664	1	0	0	0	
676	0	0	0	0	
681	0	0	0	0	
724	0	0	0	0	
771	0	0	0	0	
819	0	0	0	0	
864	0	0	0	0	
871	0	0	0	0	
1004	0	0	0	0	
1163	0	0	0	0	
1191	0	0	0	0	
1192	0	0	0	0	
1195	0	0	0	0	
1200	0	0	0	0	
1775	0	0	0	0	
1783	0	0	0	0	
1787	0	0	0	0	
2216	0	0	0	0	

2217	0	0	0	0
2226	0	0	0	0
...	...	...	...	...
87579	1	0	0	0
87655	0	0	0	0
87657	0	0	1	0
87679	0	0	0	1
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	0	0	0	0
88505	0	0	0	0
88624	0	0	0	0
88629	0	0	1	0
88630	0	0	0	0
88698	0	0	0	0
88699	0	0	0	0
88920	0	0	0	0
89080	0	0	1	1
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0

	Paper-and-Pencil	Partnerships	Pattern Building	Pattern Recognition	\
20	0	0	0	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	
251	0	0	0	0	
439	0	0	0	0	
482	0	0	0	0	
489	0	0	0	0	
491	0	0	0	0	
581	0	0	0	0	
664	0	0	0	0	
676	0	0	0	0	
681	0	0	0	0	

724	0	0	1	0
771	0	0	0	0
819	0	1	0	0
864	0	0	0	0
871	0	0	0	0
1004	0	0	0	0
1163	0	0	0	0
1191	0	0	0	0
1192	0	0	0	0
1195	0	0	0	0
1200	0	0	0	0
1775	0	0	0	0
1783	0	0	0	0
1787	0	0	0	0
2216	0	0	0	0
2217	0	0	0	0
2226	0	0	0	0
...	...	...	...	...
87579	0	0	0	0
87655	0	0	0	0
87657	0	0	0	0
87679	1	0	0	0
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	0	0	0	0
88505	0	0	0	0
88624	0	0	0	0
88629	0	0	0	0
88630	0	0	1	0
88698	0	0	0	0
88699	0	0	0	0
88920	0	0	0	0
89080	0	0	0	0
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	0	0

	Pick-up and Deliver	Player Elimination	Point to Point Movement	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	1	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	0	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	
864	0	0	0	
871	0	0	0	
1004	0	0	0	
1163	0	0	0	
1191	0	0	0	
1192	0	0	0	
1195	0	0	0	
1200	0	0	0	
1775	0	0	0	
1783	0	0	0	
1787	0	0	0	
2216	0	0	0	
2217	0	0	0	
2226	0	0	0	
...	...	...	...	
87579	0	0	0	
87655	0	0	0	
87657	0	0	0	
87679	0	0	0	
87681	1	0	0	
87714	0	0	0	
87963	0	0	0	
87985	0	0	0	
88025	0	1	0	
88028	0	1	0	
88030	0	1	0	
88031	0	1	0	
88033	0	1	0	
88034	0	1	0	
88209	0	0	0	

88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	Press Your Luck	Rock-Paper-Scissors	Role Playing	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	0	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	
864	0	0	0	
871	0	0	0	
1004	0	0	0	
1163	0	0	0	
1191	0	0	0	
1192	0	0	0	
1195	0	0	0	
1200	1	0	0	
1775	0	0	0	
1783	0	0	0	
1787	0	0	0	
2216	0	0	0	
2217	0	0	0	
2226	0	0	0	
...	...	...	...	

87579	0	0	0
87655	1	0	0
87657	0	0	0
87679	0	0	0
87681	0	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	0	0	0
88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	1	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	Roll / Spin and Move	Route/Network Building	Secret Unit Deployment	\
20	0	0	0	
50	0	0	0	
52	0	0	0	
250	0	0	0	
251	0	0	0	
439	0	0	0	
482	0	0	0	
489	0	0	0	
491	0	0	0	
581	0	0	0	
664	0	0	0	
676	0	0	0	
681	1	0	0	
724	0	0	0	
771	0	0	0	
819	0	0	0	

864	0	0	0
871	0	0	0
1004	0	0	0
1163	0	0	0
1191	0	0	0
1192	0	0	0
1195	0	0	0
1200	0	0	0
1775	0	0	0
1783	1	0	0
1787	0	0	0
2216	0	0	0
2217	1	0	0
2226	0	0	0
...	...	...	...
87579	0	0	0
87655	0	0	0
87657	0	0	0
87679	0	0	0
87681	1	0	0
87714	0	0	0
87963	0	0	0
87985	0	0	0
88025	0	0	0
88028	0	0	0
88030	0	0	0
88031	0	0	0
88033	0	0	0
88034	0	0	0
88209	0	0	0
88328	0	0	0
88500	0	0	0
88505	0	0	0
88624	0	0	0
88629	0	0	0
88630	0	0	0
88698	0	0	0
88699	0	0	0
88920	0	0	0
89080	0	0	0
89546	0	0	0
89549	0	0	0
89786	0	0	0
89788	0	0	0
89821	0	0	0

	Set Collection	Simulation	Simultaneous Action Selection	Singing	\
20	0	0	0	0	

50	0	0	0	0
52	0	0	0	0
250	0	0	0	0
251	0	0	0	0
439	0	0	0	0
482	0	0	0	0
489	0	0	0	0
491	0	0	0	0
581	0	0	0	0
664	0	0	0	0
676	0	0	0	0
681	0	0	0	0
724	0	0	0	0
771	0	0	0	0
819	0	0	0	0
864	0	0	0	0
871	0	0	0	0
1004	0	0	0	0
1163	0	0	0	0
1191	0	0	0	0
1192	0	0	0	0
1195	0	0	0	0
1200	0	0	0	0
1775	0	0	0	0
1783	0	0	0	0
1787	0	0	0	0
2216	0	0	0	0
2217	1	0	0	0
2226	0	0	0	0
...	...	...	...	...
87579	0	1	0	0
87655	0	0	0	0
87657	0	0	0	0
87679	0	0	0	0
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	1	0	0	0
88505	0	0	0	0

88624	0	0		0	0
88629	0	0		0	0
88630	0	0		0	0
88698	0	0		0	0
88699	0	0		0	0
88920	0	0		0	0
89080	0	0		0	0
89546	0	0		0	0
89549	0	0		0	0
89786	0	0		0	0
89788	0	0		0	0
89821	0	0		0	0

	Stock Holding	Storytelling	Take That	Tile Placement	Time Track	\
20	0	0	0	0	0	
50	0	0	0	0	0	
52	0	0	0	0	0	
250	0	0	0	0	0	
251	0	0	0	0	0	
439	0	0	0	0	0	
482	0	0	0	0	0	
489	0	0	0	0	0	
491	0	0	0	0	0	
581	0	0	0	1	0	
664	0	0	0	0	0	
676	0	0	0	0	0	
681	0	0	0	0	0	
724	0	0	0	0	0	
771	0	0	0	0	0	
819	0	0	0	0	0	
864	0	0	0	0	0	
871	0	0	0	0	0	
1004	0	0	0	0	0	
1163	0	0	0	0	0	
1191	0	0	0	0	0	
1192	0	0	0	0	0	
1195	0	0	0	0	0	
1200	0	0	0	0	0	
1775	0	0	0	0	0	
1783	0	0	0	0	0	
1787	0	0	0	0	0	
2216	0	0	0	1	0	
2217	0	0	0	0	0	
2226	0	0	0	0	0	
...	...	...	...	...	...	
87579	0	0	0	0	0	
87655	0	0	0	0	0	
87657	0	0	0	0	0	

87679	0	0	0	0	0
87681	0	0	0	0	1
87714	0	0	0	0	0
87963	0	0	0	0	0
87985	0	0	0	0	0
88025	0	0	0	0	0
88028	0	0	0	0	0
88030	0	0	0	0	0
88031	0	0	0	0	0
88033	0	0	0	0	0
88034	0	0	0	0	0
88209	0	0	1	0	0
88328	0	0	0	0	0
88500	0	0	0	0	0
88505	0	0	1	0	0
88624	0	0	0	0	0
88629	0	0	0	0	0
88630	0	0	0	0	0
88698	0	0	0	0	0
88699	0	0	0	0	0
88920	0	0	0	0	0
89080	0	0	0	0	0
89546	0	0	0	0	0
89549	0	0	0	0	0
89786	0	0	0	0	0
89788	0	0	0	0	0
89821	0	0	0	0	0

	Trading	Trick-taking	Variable Phase Order	Variable Player Powers	\
20	0	0	0	0	
50	0	0	0	0	
52	0	0	0	0	
250	0	0	0	0	
251	0	0	0	0	
439	0	0	0	0	
482	0	0	0	0	
489	0	0	0	0	
491	0	0	0	0	
581	0	0	0	0	
664	0	0	0	0	
676	0	0	0	0	
681	0	0	0	0	
724	0	0	0	0	
771	0	0	0	0	
819	0	0	0	0	
864	0	0	0	0	
871	0	0	0	0	
1004	0	0	0	0	

1163	0	0	0	0
1191	0	0	0	0
1192	0	1	0	0
1195	0	0	0	0
1200	0	0	0	0
1775	0	0	0	0
1783	0	0	0	0
1787	0	0	0	0
2216	0	0	0	0
2217	0	0	0	0
2226	0	0	0	0
...	...	...	...	...
87579	0	0	0	0
87655	0	0	0	0
87657	0	0	0	0
87679	0	0	0	0
87681	0	0	0	0
87714	0	0	0	0
87963	0	0	0	0
87985	0	0	0	0
88025	0	0	0	0
88028	0	0	0	0
88030	0	0	0	0
88031	0	0	0	0
88033	0	0	0	0
88034	0	0	0	0
88209	0	0	0	0
88328	0	0	0	0
88500	0	0	0	0
88505	1	0	0	0
88624	0	0	0	0
88629	0	0	0	0
88630	0	0	0	0
88698	0	0	0	0
88699	0	0	0	0
88920	0	0	0	0
89080	0	0	0	0
89546	0	0	0	0
89549	0	0	0	0
89786	0	0	0	0
89788	0	0	0	0
89821	0	0	1	1

	Voting	Worker Placement	binned.average	binary.success
20	0	0	5.0	0
50	0	0	5.0	0
52	0	0	4.0	0
250	0	0	6.0	0

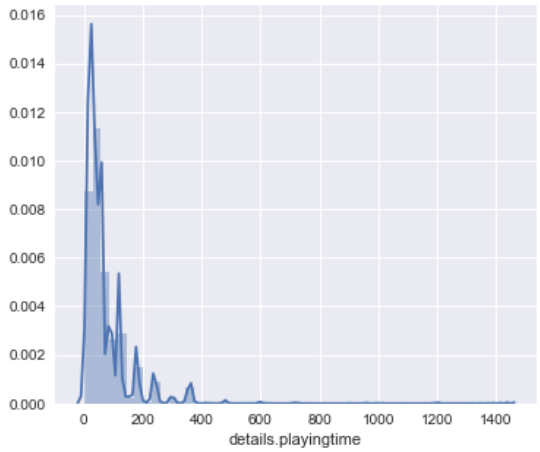
251	0	0	4.0	0
439	0	0	6.0	0
482	0	0	5.0	0
489	0	0	6.0	0
491	0	0	7.0	1
581	0	0	4.0	0
664	0	0	6.0	0
676	0	0	6.0	0
681	0	0	4.0	0
724	0	0	5.0	0
771	0	0	6.0	0
819	0	0	6.0	0
864	0	0	4.0	0
871	0	0	5.0	0
1004	0	0	2.0	0
1163	0	0	5.0	0
1191	0	0	5.0	0
1192	0	0	5.0	0
1195	0	0	5.0	0
1200	0	0	4.0	0
1775	0	0	2.0	0
1783	0	0	6.0	0
1787	0	0	5.0	0
2216	0	0	5.0	0
2217	0	0	3.0	0
2226	0	0	4.0	0
...	...	...	...	...
87579	0	0	7.0	1
87655	0	0	5.0	0
87657	0	0	6.0	0
87679	0	0	6.0	0
87681	0	0	5.0	0
87714	0	0	4.0	0
87963	0	0	6.0	0
87985	0	0	6.0	0
88025	0	0	7.0	1
88028	0	0	7.0	1
88030	0	0	8.0	1
88031	0	0	7.0	1
88033	0	0	8.0	1
88034	0	0	7.0	1
88209	0	0	6.0	0
88328	0	0	6.0	0
88500	0	0	5.0	0
88505	0	0	5.0	0
88624	0	0	7.0	1
88629	0	0	6.0	0
88630	0	0	7.0	1

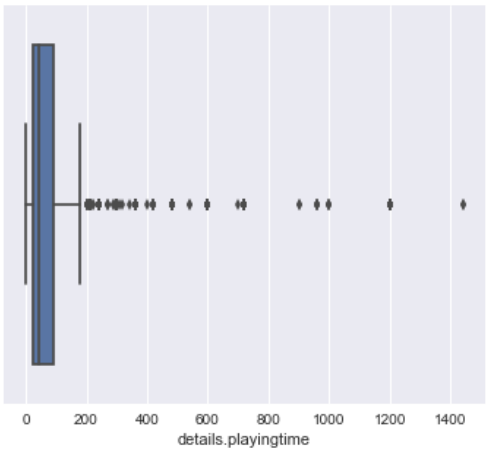
```
88698       0                0              8.0                1
88699       0                0              8.0                1
88920       0                0              5.0                0
89080       0                0              9.0                1
89546       0                0              6.0                0
89549       0                0              6.0                0
89786       0                0              7.0                1
89788       0                0              6.0                0
89821       0                0              9.0                1

[949 rows x 147 columns]
```

We observe that there are a non-trivial number of games that have a playing time of 0 minutes. We suspect that this is a data entry error and will filter these out. It is reasonable to expect that there may be some games that take one minute, but are played in rapid succession. We also observe the infamous "The Campaign for North Africa," which has been validated to being a beast of a game. Because these data might impact the models, we will limit the maximum playing time of a game to be 3 standard deviations above the mean.

For more on The Campaign for North Africa, visit: https://kotaku.com/the-notorious-board-game-that-takes-1500-hours-to-compl-1818510912

```
In [52]: df_data3['details.playingtime'].std()*3

Out[52]: 1498.1342046629006

In [53]: df_data4 = df_data3[(df_data3['details.playingtime']>=1) & (df_data3['details.playingt

In [54]: df_data3['details.playingtime'].count()-df_data4['details.playingtime'].count()

Out[54]: 975

In [55]: analyze_feature(df_data4, "details.playingtime")

count    17417.000000
mean        69.913073
std         85.272529
min          1.000000
25%         25.000000
50%         45.000000
75%         90.000000
max       1440.000000
Name: details.playingtime, dtype: float64
```



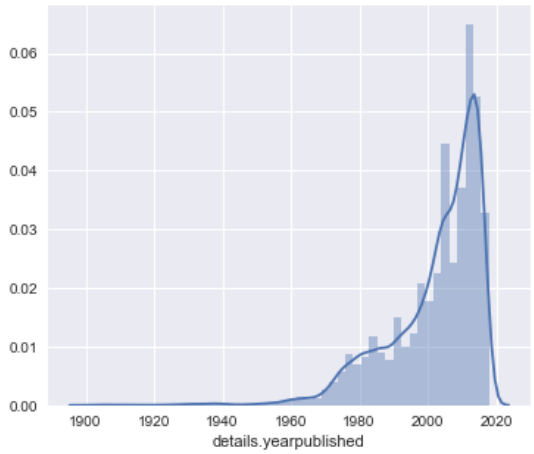


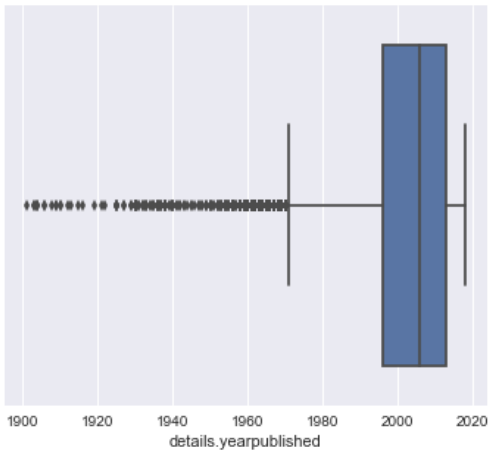
We have removed 975 games and have 17417 records in our dataframe.

Year Published

```
In [56]: analyze_feature(df_data4, "details.yearpublished")
```

```
count    17417.000000
mean      2002.295344
std         13.755996
min       1901.000000
25%       1996.000000
50%       2006.000000
75%       2013.000000
max       2018.000000
Name: details.yearpublished, dtype: float64
```





The distribution of board games appears to be skewed left, with the vast prepondarnce of the records published in the last two or so decades. No abnormalities were identified in the records.

Board Game Category

```
In [57]: #What are the first and last in the categories?
         print(cats_selector[0])
         print(cats_selector[-1])

Abstract Strategy
Zombies


In [58]: #Get the column positions
         print(df_data4.columns.get_loc('Abstract Strategy'))
         print(df_data4.columns.get_loc('Zombies')+1)

10
94


In [59]: df_data4.iloc[:, 10:94]
```

Out[59]:

	Abstract Strategy	Action / Dexterity	Adventure	Age of Reason	\
0	0	0	0	0	
1	0	0	0	0	
2	1	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	1	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	0	
13	0	0	0	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	0	0	
18	0	0	0	0	
19	0	0	0	0	
21	0	0	1	0	
22	0	0	0	0	
23	0	0	0	0	
24	0	0	0	0	
25	0	0	0	0	

26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	1	0
30	0	0	1	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	0	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	0
89570	0	0	0	0
89574	0	1	0	0
89621	0	0	0	0
89630	1	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	American Civil War	American Indian Wars	American Revolutionary War	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	

10	0	0	0
11	0	0	0
12	0	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	0	0
28	0	0	0
29	0	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	0
89517	0	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0

90206	0	0	0
90239	0	0	0
90327	0	0	0

	American West	Ancient	Animals	Arabian	Aviation / Flight	Bluffing	\
0	0	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	1	0	0	0	0	
4	0	0	0	0	0	0	
5	0	0	0	0	0	0	
6	0	0	0	0	0	0	
7	0	0	0	0	0	0	
8	0	0	0	0	0	0	
9	0	0	0	0	0	0	
10	0	0	0	0	0	0	
11	0	1	0	0	0	0	
12	0	0	0	0	0	0	
13	0	0	0	0	0	0	
14	0	0	0	0	0	1	
15	0	0	0	0	0	0	
16	0	0	0	0	0	0	
17	0	0	0	0	0	0	
18	1	0	0	0	0	1	
19	0	0	0	0	0	0	
21	0	0	0	0	0	0	
22	0	0	0	0	0	0	
23	0	0	0	0	0	0	
24	0	0	0	0	0	0	
25	0	0	0	0	0	0	
26	0	0	0	0	0	0	
27	0	0	0	0	0	0	
28	0	0	0	0	0	0	
29	0	0	0	0	0	0	
30	0	0	0	0	0	0	
...	...	...	...	...	...	...	
89496	0	0	0	0	0	0	
89497	0	0	0	0	0	0	
89498	0	0	0	0	0	0	
89502	0	0	1	0	0	0	
89505	0	0	0	0	0	0	
89517	0	0	0	0	0	0	
89521	0	0	0	0	0	0	
89526	0	0	0	0	0	1	
89536	0	0	1	0	0	0	
89537	0	0	0	0	0	0	
89547	0	0	0	0	0	0	
89570	0	0	0	0	0	0	

89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	0	0	0	0	0	1
89668	0	0	0	0	0	0
89727	0	0	0	0	0	0
89778	0	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	0	0
89962	0	0	0	0	0	0
89969	0	0	0	0	0	0
90001	0	0	0	0	0	0
90031	0	0	0	0	0	1
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	1	0	0	0
90327	0	0	0	0	0	0

	Book	Card Game	Children's Game	City Building	Civil War	\
0	0	0	0	0	0	
1	0	1	0	0	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	
5	0	0	0	0	0	
6	0	0	0	0	0	
7	0	0	0	0	0	
8	0	0	0	0	0	
9	0	0	0	0	0	
10	0	1	0	0	0	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	
16	0	0	0	0	0	
17	0	0	0	0	0	
18	0	0	0	1	0	
19	0	0	0	0	0	
21	0	0	0	0	0	
22	0	0	0	0	0	
23	0	0	0	0	0	
24	0	0	0	0	0	
25	0	0	0	0	0	
26	0	0	0	0	0	
27	0	1	0	0	0	
28	0	0	0	0	0	

29	0	0	0	0	0
30	0	0	0	0	0
...	...	...	...	...	...
89496	0	0	0	0	0
89497	0	0	0	0	0
89498	0	1	0	0	0
89502	0	0	0	0	0
89505	0	0	0	0	0
89517	0	0	0	0	0
89521	0	1	0	0	0
89526	0	1	0	0	0
89536	0	1	0	0	0
89537	0	0	0	0	0
89547	0	0	0	0	0
89570	0	0	0	0	0
89574	0	0	1	0	0
89621	0	1	0	0	0
89630	0	1	0	0	0
89668	0	0	0	0	0
89727	0	0	0	0	0
89778	0	0	0	1	0
89852	0	0	0	0	0
89856	0	1	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	1	0	0	0
90001	0	0	0	1	0
90031	0	1	0	0	0
90065	0	1	0	1	0
90167	0	0	0	0	0
90206	0	1	0	0	0
90239	0	0	0	0	0
90327	0	1	0	0	0

	Civilization	Collectible Components	Comic Book / Strip	Deduction	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	1	0	0	0	
6	0	0	0	0	
7	1	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	0	

13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	1	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	1	0	0	0
24	0	0	0	0
25	1	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	1	0	0	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	1	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	1
89778	0	0	0	0
89852	1	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Dice	Economic	Educational	Electronic	Environmental	\
0	0	1	0	0	0	
1	0	0	0	0	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	1	0	0	0	
5	0	0	0	0	0	
6	0	0	0	0	0	
7	0	0	0	0	0	
8	0	0	0	0	0	
9	0	0	0	0	0	
10	0	0	0	0	0	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	1	0	0	0	
16	1	0	0	0	0	
17	0	0	0	0	0	
18	0	0	0	0	0	
19	0	0	0	0	0	
21	0	0	0	0	0	
22	0	0	0	0	0	
23	0	0	0	0	0	
24	0	0	0	0	0	
25	0	1	0	0	0	
26	0	1	0	0	0	
27	0	0	0	0	0	
28	0	0	0	0	0	
29	0	0	0	1	0	
30	0	0	0	0	0	
...	...	...	...	...	...	
89496	1	0	0	0	0	
89497	1	0	0	0	0	
89498	0	0	0	0	0	
89502	0	0	0	0	0	
89505	0	0	0	0	0	
89517	0	0	0	0	0	
89521	0	0	0	0	0	
89526	0	0	0	0	0	
89536	0	0	0	0	0	
89537	0	0	0	0	0	
89547	1	0	0	0	0	
89570	0	0	0	0	0	
89574	0	0	0	0	0	
89621	0	0	0	0	0	
89630	0	1	0	0	0	

89668	0	0	0	0	0
89727	0	0	0	0	0
89778	1	1	0	0	0
89852	0	1	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	1	0	0	0	0
90001	0	0	0	0	0
90031	0	1	0	0	0
90065	0	1	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	0	0
90327	0	0	0	0	0

	Expansion for Base-game	Exploration	Fan Expansion	Fantasy	Farming	\
0	0	0	0	0	0	
1	0	0	0	1	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	
5	0	0	0	0	0	
6	0	0	0	0	0	
7	0	0	0	1	0	
8	0	1	0	0	0	
9	0	0	0	1	0	
10	0	0	0	0	1	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	
16	0	0	0	0	0	
17	0	0	0	0	0	
18	0	0	0	0	0	
19	0	0	0	0	0	
21	0	1	0	1	0	
22	0	0	0	1	0	
23	0	0	0	0	0	
24	0	1	0	1	0	
25	0	0	0	0	0	
26	0	0	0	0	0	
27	0	0	0	0	0	
28	0	0	0	0	0	
29	0	1	0	1	0	
30	0	1	0	1	0	
...	...	...	...	...	...	

89496	0	0	0	0	0
89497	0	0	0	0	0
89498	0	0	0	0	0
89502	0	0	0	1	0
89505	0	0	0	0	0
89517	0	0	0	1	0
89521	0	0	0	0	0
89526	0	0	0	0	0
89536	0	0	0	0	0
89537	0	0	0	0	0
89547	0	0	0	0	0
89570	0	0	0	1	0
89574	0	0	0	0	0
89621	0	0	0	0	0
89630	0	0	0	0	0
89668	0	1	0	1	0
89727	0	0	0	0	0
89778	0	0	0	1	0
89852	0	1	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	0	0	0	0
90001	0	0	0	0	0
90031	0	0	0	0	0
90065	0	0	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	0	0
90327	0	0	0	1	0

	Fighting	Game System	Horror	Humor	Industry / Manufacturing	\
0	0	0	0	0	0	
1	0	0	0	0	0	
2	0	0	0	0	0	
3	0	0	0	0	0	
4	0	0	0	0	0	
5	0	0	0	0	0	
6	0	0	0	0	0	
7	0	0	0	0	0	
8	0	0	0	0	0	
9	0	0	0	0	0	
10	0	0	0	0	0	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	

16	1	0	0	0	0
17	0	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	1	0
28	1	0	0	1	0
29	1	0	0	0	0
30	1	0	0	0	0
...	...	...	...	...	...
89496	0	0	0	0	0
89497	0	0	0	0	0
89498	0	0	0	0	0
89502	0	0	0	0	0
89505	0	0	0	0	0
89517	0	0	0	0	0
89521	0	0	0	0	0
89526	1	0	0	0	0
89536	0	0	0	0	0
89537	0	0	0	0	0
89547	1	0	0	0	0
89570	0	0	0	0	0
89574	0	0	0	0	0
89621	0	0	0	0	0
89630	0	0	0	0	1
89668	0	0	0	0	0
89727	0	0	0	0	0
89778	0	0	0	0	0
89852	0	0	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	1	0
89969	0	0	0	1	0
90001	0	0	0	0	0
90031	0	0	0	0	0
90065	0	0	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	0	0
90327	0	0	0	1	0

	Korean War	Mafia	Math	Mature / Adult	Maze	Medical	Medieval	\
0	0	0	0	0	0	0	0	

1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0
25	0	0	0	0	0	0	1
26	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...
89496	0	0	0	0	0	0	0
89497	0	0	0	0	0	0	0
89498	0	0	0	0	0	0	0
89502	0	0	0	0	0	0	0
89505	0	0	0	0	0	0	0
89517	0	0	0	0	0	0	1
89521	0	0	0	0	0	0	0
89526	0	0	0	0	0	0	1
89536	0	0	0	0	0	0	0
89537	0	0	0	0	0	0	0
89547	0	0	0	0	0	0	0
89570	0	0	0	0	0	0	0
89574	0	0	0	0	0	0	0
89621	0	0	0	0	0	0	0
89630	0	0	1	0	0	0	0
89668	0	0	0	0	0	0	0
89727	0	0	0	0	0	0	0
89778	0	0	0	0	0	0	0

89852	0	0	0	0	0	0	0
89856	0	0	0	0	0	0	0
89881	0	0	0	0	0	0	0
89962	0	0	0	0	0	0	0
89969	0	0	0	0	0	0	0
90001	0	0	0	0	0	0	0
90031	0	0	0	0	0	0	0
90065	0	0	0	0	0	0	0
90167	0	0	0	0	0	0	0
90206	0	0	0	0	0	0	0
90239	0	0	0	0	0	0	0
90327	0	0	0	0	0	0	1

	Memory_category	Miniatures	Modern Warfare	Movies / TV / Radio theme	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	0	
13	0	0	0	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	1	0	0	
18	0	0	0	0	
19	0	0	0	0	
21	0	0	0	0	
22	0	0	0	0	
23	0	0	0	0	
24	0	0	0	0	
25	0	0	0	0	
26	0	0	0	0	
27	0	0	0	0	
28	0	0	0	0	
29	0	0	0	0	
30	0	1	0	0	
...	...	...	...	...	
89496	0	0	0	0	
89497	0	0	0	0	
89498	0	0	0	0	

89502	0	1	0	0
89505	0	0	0	0
89517	0	0	0	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	1
89570	0	1	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Murder/Mystery	Music	Mythology	Napoleonic	Nautical	Negotiation	\
0	0	0	0	0	0	1	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	
5	0	0	0	0	1	0	
6	0	0	0	0	0	0	
7	0	0	0	0	0	0	
8	0	0	0	0	0	0	
9	0	0	0	0	0	0	
10	0	0	0	0	0	1	
11	0	0	1	0	0	0	
12	0	0	0	0	0	1	
13	0	0	0	0	0	1	
14	0	0	0	0	0	1	
15	0	0	0	0	0	0	
16	0	0	0	0	0	0	
17	0	0	0	0	0	0	
18	0	0	0	0	0	0	

19	0	0	0	0	0	0
21	0	0	0	0	0	0
22	0	0	0	0	0	0
23	0	0	0	0	0	1
24	0	0	0	0	0	0
25	0	0	0	0	0	0
26	0	0	0	0	0	0
27	0	0	0	0	0	1
28	0	0	0	0	0	0
29	0	0	0	0	0	0
30	0	0	0	0	0	0
...	...	...	...	...	...	...
89496	0	0	0	0	0	0
89497	0	0	0	0	0	0
89498	0	0	0	0	0	0
89502	0	0	0	0	0	0
89505	0	0	0	0	0	0
89517	0	0	0	0	0	0
89521	0	0	0	0	0	0
89526	0	0	0	0	0	0
89536	0	0	0	0	0	0
89537	0	0	0	0	0	0
89547	0	0	0	0	0	0
89570	0	0	0	0	0	0
89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	0	0	0	0	1	0
89668	0	0	0	0	0	0
89727	0	0	0	0	0	0
89778	0	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	0	0
89962	0	0	0	0	0	0
89969	0	0	0	0	0	0
90001	0	0	0	0	0	0
90031	0	0	0	0	0	0
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	0	0	0
90327	0	0	1	0	0	0

	Novel-based	Number	Party Game	Pike and Shot	Pirates	Political	\
0	0	0	0	0	0	1	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	

4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0
14	0	0	0	0	0	0
15	0	0	0	0	0	0
16	0	0	0	0	0	0
17	0	0	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	0	0
21	0	0	0	0	0	0
22	0	0	0	0	0	1
23	0	0	0	0	0	1
24	0	0	0	0	0	0
25	0	0	0	0	0	0
26	0	0	0	0	0	1
27	0	0	0	0	0	1
28	0	0	0	0	0	0
29	0	0	0	0	0	0
30	0	0	0	0	0	0
...	...	...	...	...	...	...
89496	0	0	0	0	0	0
89497	0	0	0	0	0	0
89498	0	0	0	0	0	0
89502	0	0	0	0	0	0
89505	0	0	0	0	0	0
89517	0	0	0	0	0	0
89521	0	0	0	0	0	0
89526	0	0	0	0	0	0
89536	0	0	0	0	0	0
89537	0	1	0	0	0	0
89547	0	0	0	0	0	0
89570	0	0	0	0	0	0
89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	0	0	0	0	1	0
89668	0	0	0	0	0	0
89727	0	0	1	0	0	0
89778	0	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	1	0
89881	0	0	0	0	0	0

89962	0	0	1	0	0	0
89969	0	0	0	0	0	0
90001	0	0	0	0	0	0
90031	0	0	0	0	0	0
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	0	0	0
90327	0	0	0	0	0	0

	Post-Napoleonic	Prehistoric	Print & Play	Puzzle	Racing	Real-time	\
0	0	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	
5	0	0	0	0	0	0	
6	0	0	0	0	0	0	
7	0	0	0	0	0	0	
8	0	0	0	0	0	0	
9	0	0	0	0	0	0	
10	0	0	0	0	0	0	
11	0	0	0	0	0	0	
12	0	0	0	0	0	0	
13	0	0	0	0	0	0	
14	0	0	0	0	0	0	
15	0	0	0	0	0	0	
16	0	0	1	0	0	0	
17	0	0	0	0	1	0	
18	0	0	0	0	0	0	
19	0	0	0	0	0	0	
21	0	0	0	0	0	0	
22	0	0	0	0	0	0	
23	0	0	0	0	0	0	
24	0	0	0	0	0	0	
25	0	0	0	0	0	0	
26	0	0	0	0	0	0	
27	0	0	0	0	0	0	
28	0	0	0	0	0	0	
29	0	0	0	0	0	0	
30	0	0	0	0	0	0	
...	...	...	...	...	...	...	
89496	0	0	0	0	0	0	
89497	0	0	0	0	0	0	
89498	0	0	0	0	0	0	
89502	0	0	0	0	0	0	
89505	0	0	0	0	1	0	
89517	0	0	0	0	0	0	

89521	0	0	0	0	0	0
89526	0	0	0	0	0	0
89536	0	0	0	0	0	0
89537	0	0	0	1	0	0
89547	0	0	0	0	0	0
89570	0	0	0	0	0	0
89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	0	0	0	0	0	0
89668	0	0	0	0	0	0
89727	0	0	0	0	0	0
89778	0	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	1	0
89962	0	0	0	0	0	0
89969	0	0	0	0	0	0
90001	0	0	0	0	0	0
90031	0	0	0	0	0	0
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	1	0	0
90327	0	0	0	0	0	0

	Religious	Renaissance	Science Fiction	Space Exploration	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	0	
13	0	0	0	0	
14	0	0	1	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	1	0	
18	0	0	0	0	
19	0	0	1	0	
21	0	0	0	0	
22	0	0	0	0	

23	0	0	1	1
24	0	0	0	0
25	0	1	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	0	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	1	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	1	1
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	1	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Spies/Secret Agents	Sports	Territory Building	Trains	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	

7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	1	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0

90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Transportation	Travel	Trivia	Video Game Theme	Vietnam War	Wargame	\
0	0	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	
5	0	0	0	0	0	0	
6	0	0	0	0	0	0	
7	0	0	0	0	0	0	
8	0	0	0	0	0	0	
9	0	1	0	0	0	0	
10	0	0	0	0	0	0	
11	0	0	0	0	0	0	
12	0	0	0	0	0	0	
13	0	0	0	0	0	0	
14	0	0	0	0	0	0	
15	0	0	0	0	0	0	
16	0	0	0	0	0	0	
17	0	0	0	0	0	0	
18	0	0	0	0	0	0	
19	0	0	0	0	0	0	
21	0	0	0	0	0	0	
22	0	0	0	0	0	1	
23	0	0	0	0	0	1	
24	0	0	0	0	0	1	
25	0	0	0	0	0	0	
26	0	0	0	0	0	1	
27	0	0	0	0	0	0	
28	0	0	0	0	0	0	
29	0	0	0	0	0	0	
30	0	0	0	0	0	0	
...	...	...	...	...	...	...	
89496	0	0	0	0	0	0	
89497	0	0	0	0	0	0	
89498	0	0	0	0	0	0	
89502	0	0	0	0	0	0	
89505	0	0	0	0	0	0	
89517	0	0	0	0	0	0	
89521	0	0	0	0	0	0	
89526	0	0	0	0	0	0	
89536	0	0	0	0	0	0	

89537	0	0	0	0	0	0
89547	0	0	0	0	0	0
89570	0	0	0	0	0	0
89574	0	0	0	0	0	0
89621	0	0	0	0	0	0
89630	1	0	0	0	0	0
89668	0	0	0	0	0	0
89727	0	0	0	0	0	0
89778	0	0	0	0	0	0
89852	0	0	0	0	0	0
89856	0	0	0	0	0	0
89881	0	0	0	0	0	0
89962	0	0	0	0	0	0
89969	0	0	0	0	0	0
90001	0	0	0	0	0	0
90031	0	0	0	0	0	0
90065	0	0	0	0	0	0
90167	0	0	0	0	0	0
90206	0	0	0	0	0	0
90239	0	0	0	0	0	0
90327	0	0	0	0	0	0

	Word Game	World War I	World War II	Zombies
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0

```
26              0            0              0        0
27              0            0              0        0
28              0            0              0        0
29              0            0              0        0
30              0            0              0        0
...           ...          ...            ...      ...
89496           0            0              0        0
89497           0            0              0        0
89498           0            0              0        0
89502           0            0              0        0
89505           0            0              0        0
89517           0            0              0        0
89521           0            0              0        0
89526           0            0              0        0
89536           0            0              0        0
89537           0            0              0        0
89547           0            0              0        0
89570           0            0              0        0
89574           0            0              0        0
89621           0            0              0        0
89630           0            0              0        0
89668           0            0              0        0
89727           0            0              0        0
89778           0            0              0        0
89852           0            0              0        0
89856           0            0              0        0
89881           0            0              0        0
89962           0            0              0        0
89969           0            0              0        0
90001           0            0              0        0
90031           0            0              0        0
90065           0            0              0        0
90167           0            0              0        0
90206           0            0              0        0
90239           0            0              0        0
90327           0            0              0        0

[17417 rows x 84 columns]
```

```
In [60]: s_cat_overivew = df_data4.iloc[:, 10:94].sum()

In [61]: df_cat_overivew = pd.Series.to_frame(s_cat_overivew)

In [62]: df_cat_overivew[0].describe()

Out[62]: count      84.000000
         mean      529.452381
         std       702.365820
         min         2.000000
```

```
25%       149.500000
50%       278.000000
75%       637.250000
max      4809.000000
Name: 0, dtype: float64
```

We see that across all the categories, the minimum number of board games per one category is two, and the maximum is 4809, with an average of approximately 529 to each category.

Board Game Mechanic

```
In [63]: #What are the first and last in the categories?
         print(mechs_selector[0])
         print(mechs_selector[-1])
```

```
Acting
Worker Placement
```

```
In [64]: #Get the column positions
         print(df_data4.columns.get_loc('Acting'))
         print(df_data4.columns.get_loc('Worker Placement')+1)
```

```
94
145
```

```
In [65]: df_data4.iloc[:, 94:145]
```

```
Out[65]:        Acting  Action / Movement Programming  Action Point Allowance System  \
         0           0                              0                              0
         1           0                              0                              0
         2           0                              0                              0
         3           0                              0                              1
         4           0                              0                              0
         5           0                              0                              0
         6           0                              0                              0
         7           0                              0                              0
         8           0                              0                              0
         9           0                              0                              0
         10          0                              0                              0
         11          0                              0                              0
         12          0                              0                              0
         13          0                              0                              0
         14          0                              0                              0
         15          0                              0                              0
         16          0                              0                              0
         17          0                              1                              0
         18          0                              0                              0
```

19	0	0	1
21	0	1	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	0	0
28	0	0	1
29	0	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	0
89517	0	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	1
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Area Control / Area Influence	Area Enclosure	Area Movement	\
0	1	0	0	
1	0	0	0	
2	1	0	0	
3	1	0	0	

4	0	0	0
5	0	0	0
6	0	1	0
7	0	0	0
8	1	0	0
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	1
26	0	0	0
27	0	0	0
28	0	0	0
29	0	0	1
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	1
89517	1	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	1	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	1	0	0
89856	0	0	0
89881	0	0	0

89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Area-Impulse	Auction/Bidding	Betting/Wagering	\
0	0	1	0	
1	0	0	0	
2	0	0	0	
3	0	1	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	
10	0	0	0	
11	0	1	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	1	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	1	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	0	
30	0	0	0	
...	...	...	...	
89496	0	0	0	
89497	0	0	0	
89498	0	0	0	
89502	0	0	0	
89505	0	0	0	
89517	0	0	0	

89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	1	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	1
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Campaign / Battle Card Driven	Card Drafting	Chit-Pull System	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	1	0	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
21	0	0	0	
22	0	0	0	

23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	1	0
28	0	0	0
29	0	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	1	0
89517	0	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	1	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	1	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Co-operative Play	Commodity Speculation	Crayon Rail System	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	

7	0	0	0
8	0	0	0
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	1	0
27	0	0	0
28	0	0	0
29	0	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	0
89517	0	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	1	0	0
89574	0	0	0
89621	0	0	0
89630	0	1	0
89668	0	0	0
89727	1	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0

90031	0	1	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Deck / Pool Building	Dice Rolling	Grid Movement	Hand Management	\
0	0	1	0	1	
1	0	0	0	0	
2	0	0	0	1	
3	0	0	0	0	
4	0	0	0	1	
5	0	1	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	1	
10	0	0	0	1	
11	0	0	0	0	
12	0	1	0	1	
13	0	0	0	0	
14	0	0	0	1	
15	0	0	0	0	
16	0	1	0	0	
17	0	0	1	0	
18	0	0	0	0	
19	0	0	0	0	
21	0	0	0	0	
22	0	1	0	0	
23	0	1	0	0	
24	0	0	0	0	
25	0	0	0	0	
26	0	1	0	0	
27	0	1	0	0	
28	0	0	0	0	
29	0	0	0	0	
30	0	1	1	0	
...	...	...	...	...	
89496	0	0	0	0	
89497	0	1	0	0	
89498	0	0	0	0	
89502	0	0	1	0	
89505	1	0	1	0	
89517	0	1	0	1	
89521	0	0	0	1	
89526	0	0	0	1	
89536	0	0	0	0	

89537	0	0	0	0
89547	1	1	0	0
89570	0	1	0	0
89574	0	0	0	0
89621	0	0	0	1
89630	0	0	0	1
89668	0	1	0	0
89727	0	0	0	0
89778	0	1	0	0
89852	0	1	0	1
89856	0	0	0	1
89881	0	0	0	0
89962	0	0	0	0
89969	0	1	0	0
90001	1	0	0	0
90031	0	0	0	0
90065	0	0	0	1
90167	0	0	0	0
90206	0	0	0	1
90239	0	0	0	0
90327	1	0	0	1

	Hex-and-Counter	Line Drawing	Memory_mechanics	Modular Board	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	1	
8	0	0	0	0	
9	0	0	0	0	
10	0	0	0	0	
11	0	0	0	0	
12	0	0	0	1	
13	0	0	0	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	0	1	
18	0	0	0	0	
19	0	0	0	0	
21	0	0	0	1	
22	1	0	0	0	
23	1	0	0	1	
24	0	0	0	0	
25	0	0	0	0	

26	0	0	0	0
27	0	0	0	0
28	0	0	0	1
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	0	0
89521	0	0	0	0
89526	0	0	1	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	0
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	0	1
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Paper-and-Pencil	Partnerships	Pattern Building	Pattern Recognition	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	1	1	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	

10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
13	0	0	0	0
14	0	0	0	0
15	0	0	0	0
16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
21	0	0	0	0
22	0	0	0	0
23	0	0	0	0
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	0
28	0	0	0	0
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	0	0
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	0
89570	0	0	0	0
89574	0	0	1	0
89621	0	0	0	0
89630	0	0	0	0
89668	0	0	0	0
89727	0	0	0	0
89778	0	0	0	0
89852	0	0	0	0
89856	0	1	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0

90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Pick-up and Deliver	Player Elimination	Point to Point Movement	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	1	
10	0	0	0	
11	0	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	0	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
21	0	0	0	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	0	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	
29	0	0	0	
30	0	0	0	
...	...	...	...	
89496	0	0	0	
89497	0	0	0	
89498	0	0	0	
89502	0	0	0	
89505	0	0	0	
89517	0	0	0	
89521	0	0	0	
89526	0	0	0	
89536	0	0	0	
89537	0	0	0	
89547	0	0	0	
89570	0	0	0	

89574	0	0	0
89621	0	0	0
89630	1	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Press Your Luck	Rock-Paper-Scissors	Role Playing	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	0	0	
10	0	0	0	
11	1	0	0	
12	0	0	0	
13	0	0	0	
14	0	0	0	
15	0	0	0	
16	1	0	0	
17	0	0	0	
18	0	0	0	
19	0	0	0	
21	0	1	1	
22	0	0	0	
23	0	0	0	
24	0	0	0	
25	0	0	0	
26	0	0	0	
27	0	0	0	
28	0	0	0	

29	1	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	1	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	0
89517	0	0	0
89521	0	0	0
89526	0	0	0
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	1	0	1
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Roll / Spin and Move	Route/Network Building	Secret Unit Deployment	\
0	0	0	0	
1	0	0	0	
2	0	0	0	
3	0	0	0	
4	0	0	0	
5	0	0	0	
6	0	0	0	
7	0	0	0	
8	0	0	0	
9	0	1	0	
10	0	0	0	
11	0	0	0	
12	0	1	0	

13	1	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
21	0	0	0
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	1	0
28	0	0	0
29	0	0	0
30	0	0	0
...	...	...	...
89496	0	0	0
89497	0	0	0
89498	0	0	0
89502	0	0	0
89505	0	0	0
89517	0	0	0
89521	0	0	0
89526	0	0	1
89536	0	0	0
89537	0	0	0
89547	0	0	0
89570	0	0	0
89574	0	0	0
89621	0	0	0
89630	0	0	0
89668	0	0	0
89727	0	0	0
89778	0	0	0
89852	0	0	0
89856	0	0	0
89881	0	0	0
89962	0	0	0
89969	0	0	0
90001	0	0	0
90031	0	0	0
90065	0	0	0
90167	0	0	0
90206	0	0	0
90239	0	0	0
90327	0	0	0

	Set Collection	Simulation	Simultaneous Action Selection	Singing	\
0	0	0	1	0	
1	0	0	0	0	
2	1	0	0	0	
3	1	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	1	0	0	0	
11	1	0	0	0	
12	0	0	0	0	
13	1	0	1	0	
14	0	0	0	0	
15	0	0	0	0	
16	0	0	0	0	
17	0	0	1	0	
18	0	0	0	0	
19	0	0	0	0	
21	0	0	1	0	
22	0	0	0	0	
23	0	0	0	0	
24	0	0	0	0	
25	0	0	0	0	
26	0	0	0	0	
27	0	0	0	0	
28	0	0	0	0	
29	0	0	0	0	
30	0	0	0	0	
...	...	...	...	...	
89496	0	0	0	0	
89497	0	0	0	0	
89498	0	0	0	0	
89502	1	0	0	0	
89505	0	0	0	0	
89517	0	0	0	0	
89521	0	0	0	0	
89526	0	0	0	0	
89536	0	0	0	0	
89537	0	0	1	0	
89547	0	0	0	0	
89570	0	0	0	0	
89574	0	0	0	0	
89621	1	0	0	0	
89630	0	0	0	0	

89668	0	0	0	0
89727	0	0	0	0
89778	1	0	0	0
89852	0	0	0	0
89856	0	0	0	0
89881	0	0	0	0
89962	0	0	0	0
89969	1	0	0	0
90001	0	0	0	0
90031	0	0	1	0
90065	0	0	0	0
90167	0	0	0	0
90206	1	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Stock Holding	Storytelling	Take That	Tile Placement	Time Track	\
0	0	0	0	0	0	
1	0	0	0	0	0	
2	0	0	0	1	0	
3	0	0	0	0	0	
4	1	0	0	1	0	
5	0	0	0	0	0	
6	0	0	0	1	0	
7	0	0	0	0	0	
8	0	0	0	1	0	
9	0	0	0	0	0	
10	0	0	0	0	0	
11	0	0	0	0	0	
12	0	0	0	0	0	
13	0	0	0	0	0	
14	0	0	0	0	0	
15	0	0	0	0	0	
16	0	0	0	0	0	
17	0	0	0	0	0	
18	0	0	0	1	0	
19	0	0	0	0	0	
21	0	0	0	0	0	
22	0	0	0	0	0	
23	0	0	0	1	0	
24	0	0	0	0	0	
25	0	0	0	0	0	
26	0	0	0	0	0	
27	0	0	0	1	0	
28	0	0	0	0	0	
29	0	0	0	0	0	
30	0	0	0	0	0	
...	...	...	...	...	...	

89496	0	0	0	0	0
89497	0	0	0	0	0
89498	0	0	0	1	0
89502	0	0	0	0	0
89505	0	0	0	0	0
89517	0	0	0	0	0
89521	0	0	0	0	0
89526	0	0	1	0	0
89536	0	0	0	0	0
89537	0	0	0	1	0
89547	0	0	0	0	0
89570	0	0	0	0	0
89574	0	0	0	0	0
89621	0	0	1	0	0
89630	1	0	1	0	0
89668	0	1	0	1	0
89727	0	0	0	0	0
89778	0	0	0	0	0
89852	0	0	0	0	0
89856	0	0	0	0	0
89881	0	0	0	0	0
89962	0	0	0	0	0
89969	0	0	0	0	0
90001	0	0	0	0	0
90031	0	0	0	0	0
90065	0	0	0	0	0
90167	0	0	0	0	0
90206	0	0	0	0	0
90239	0	0	0	1	0
90327	0	0	0	0	0

	Trading	Trick-taking	Variable Phase Order	Variable Player Powers	\
0	0	0	0	0	
1	0	1	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	
5	0	0	0	0	
6	0	0	0	0	
7	0	0	0	0	
8	0	0	0	0	
9	0	0	0	0	
10	1	0	0	0	
11	0	0	0	0	
12	1	0	0	0	
13	0	0	0	0	
14	0	0	0	1	
15	0	0	0	0	

16	0	0	0	0
17	0	0	0	0
18	0	0	0	0
19	0	0	0	0
21	0	0	0	1
22	0	0	1	0
23	0	0	0	1
24	0	0	0	0
25	0	0	0	0
26	0	0	0	0
27	0	0	0	1
28	0	0	0	0
29	0	0	0	0
30	0	0	0	0
...	...	...	...	...
89496	0	0	0	0
89497	0	0	0	0
89498	0	0	0	0
89502	0	0	0	0
89505	0	0	0	0
89517	0	0	0	1
89521	0	0	0	0
89526	0	0	0	0
89536	0	0	0	0
89537	0	0	0	0
89547	0	0	0	1
89570	0	0	0	0
89574	0	0	0	0
89621	0	0	0	0
89630	0	0	0	1
89668	0	0	0	1
89727	0	0	0	0
89778	0	0	0	1
89852	0	0	0	1
89856	0	0	0	0
89881	0	0	0	1
89962	0	0	0	0
89969	0	0	0	0
90001	0	0	0	0
90031	0	0	0	0
90065	0	0	0	0
90167	0	0	0	0
90206	0	0	0	0
90239	0	0	0	0
90327	0	0	0	0

	Voting	Worker Placement
0	0	0

1	0	0
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0
18	1	0
19	0	0
21	0	0
22	0	0
23	1	0
24	0	0
25	0	0
26	0	0
27	0	0
28	0	0
29	0	0
30	0	0
...	...	...
89496	0	0
89497	0	0
89498	0	0
89502	0	0
89505	0	0
89517	0	0
89521	0	0
89526	0	0
89536	0	0
89537	0	0
89547	0	0
89570	0	0
89574	0	0
89621	0	0
89630	0	0
89668	0	0
89727	0	0
89778	0	1

```
89852       0                 0
89856       1                 0
89881       0                 0
89962       0                 0
89969       0                 0
90001       0                 0
90031       0                 0
90065       0                 1
90167       0                 0
90206       0                 0
90239       0                 0
90327       0                 0

[17417 rows x 51 columns]
```

```
In [66]: s_mechs_overivew = df_data4.iloc[:, 94:145].sum()

In [67]: df_mechs_overview = pd.Series.to_frame(s_mechs_overivew)

In [68]: df_mechs_overview
```

Out[68]:

	0
Acting	189
Action / Movement Programming	244
Action Point Allowance System	805
Area Control / Area Influence	1044
Area Enclosure	174
Area Movement	883
Area-Impulse	72
Auction/Bidding	965
Betting/Wagering	319
Campaign / Battle Card Driven	376
Card Drafting	1291
Chit-Pull System	165
Co-operative Play	813
Commodity Speculation	222
Crayon Rail System	26
Deck / Pool Building	429
Dice Rolling	3718
Grid Movement	576
Hand Management	3081
Hex-and-Counter	1962
Line Drawing	88
Memory_mechanics	823
Modular Board	1213
Paper-and-Pencil	201
Partnerships	648
Pattern Building	599
Pattern Recognition	480

```
Pick-up and Deliver             510
Player Elimination              341
Point to Point Movement         717
Press Your Luck                 504
Rock-Paper-Scissors             141
Role Playing                    380
Roll / Spin and Move           1524
Route/Network Building          394
Secret Unit Deployment          461
Set Collection                 2114
Simulation                      891
Simultaneous Action Selection   947
Singing                          47
Stock Holding                   280
Storytelling                    289
Take That                       443
Tile Placement                 1385
Time Track                       75
Trading                         612
Trick-taking                    304
Variable Phase Order            256
Variable Player Powers         1621
Voting                          342
Worker Placement                450
```

```
In [69]: df_mechs_overview[0].describe()
```

```
Out[69]: count      51.000000
         mean      714.392157
         std       734.973607
         min        26.000000
         25%       268.000000
         50%       461.000000
         75%       887.000000
         max      3718.000000
         Name: 0, dtype: float64
```

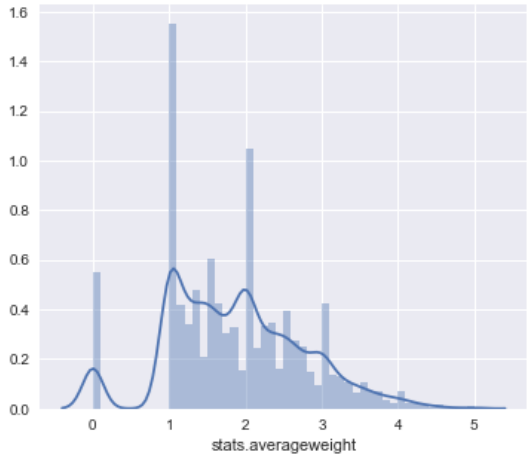
We see that across all the mechanics, the minimum number of board games per one mechanic is is 26, and the maximum is 3718, with an average of approximately 714 to each mechanic.

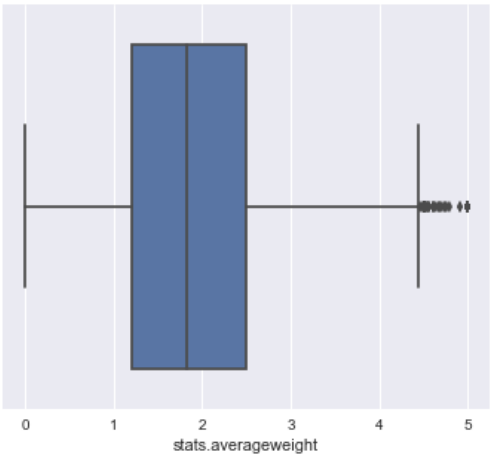
Average Weight

```
In [70]: analyze_feature(df_data4, "stats.averageweight")
```

```
count    17417.000000
mean         1.873099
std          0.899252
min          0.000000
25%          1.200000
```

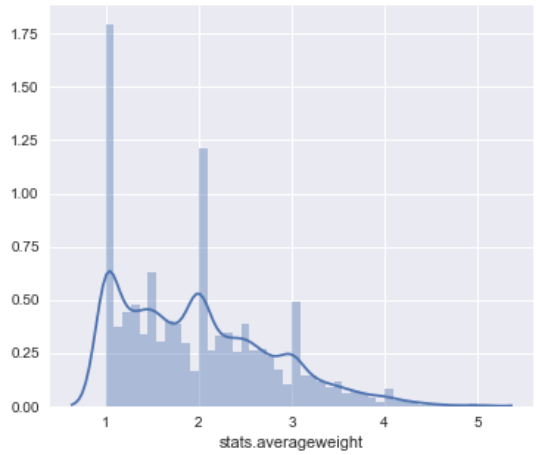
```
50%          1.833300
75%          2.500000
max          5.000000
Name: stats.averageweight, dtype: float64
```

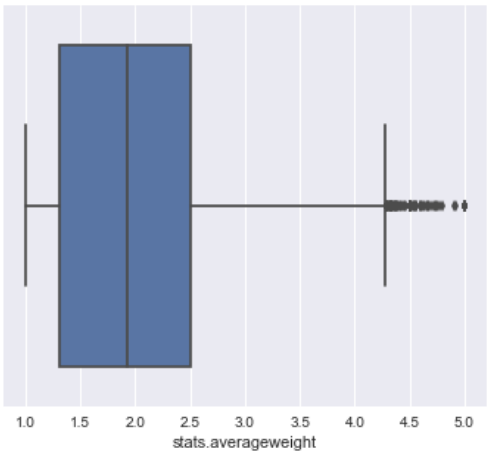




BoardGameGeek has five categorical variables for "weight" which answers the question: "How heavy (difficult/complex) is this game?" Light = 1, Medium Light = 2, Medium = 3, Medium Heavy = 4, and Heavy = 5. Consequently, any board games with a weight of less than 1 should be eliminated, and anything above 5 should be eliminated as well.

```
In [71]: df_data5 = df_data4[(df_data4['stats.averageweight']>=1)]

In [72]: df_data4['stats.averageweight'].count()-df_data5['stats.averageweight'].count()

Out[72]: 956

In [73]: analyze_feature(df_data5, "stats.averageweight")

count    16461.000000
mean         1.981882
std          0.800007
min          1.000000
25%          1.318200
50%          1.933300
75%          2.500000
max          5.000000
Name: stats.averageweight, dtype: float64
```



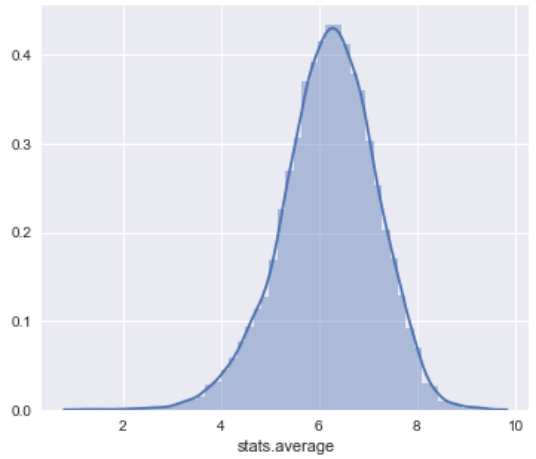


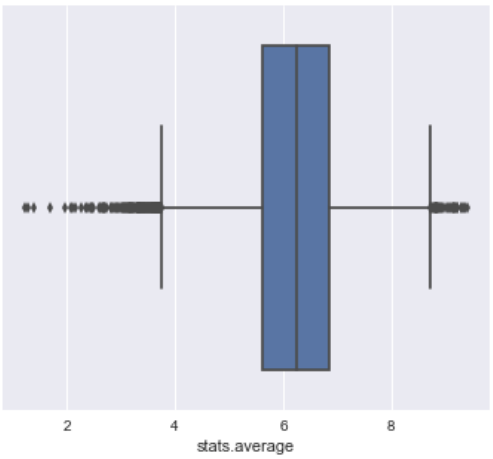
We removed 956 games from our record and now have 16461.

Average Score

```
In [74]: analyze_feature(df_data5, "stats.average")
```

```
count    16461.000000
mean         6.200449
std          0.959954
min          1.226530
25%          5.609620
50%          6.241110
75%          6.855570
max          9.411770
Name: stats.average, dtype: float64
```



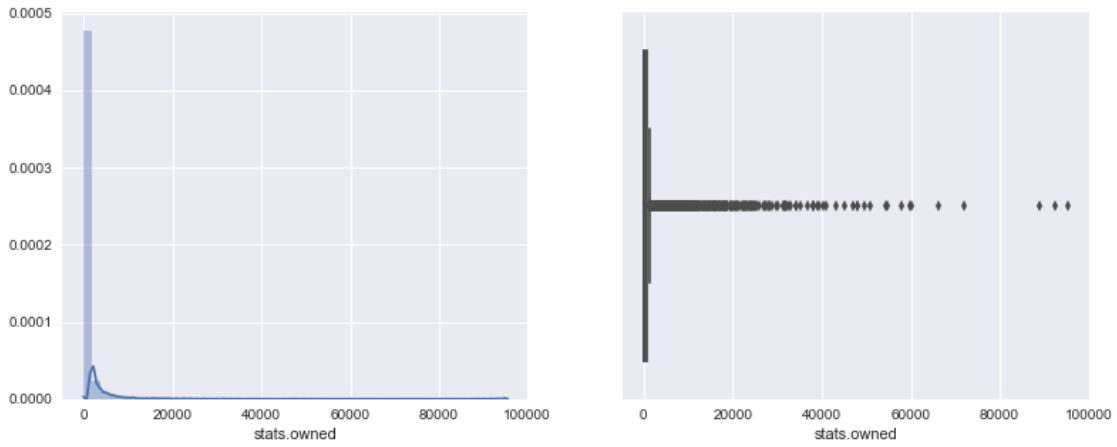


Scores can range from 1 to 10, and from investigating the data, we see a small left-skew, but no data to be concerned about.

Average Owned

```
In [75]: analyze_feature(df_data5, "stats.owned")
```

```
count    16461.000000
mean       882.566065
std       3172.514864
min          1.000000
25%         67.000000
50%        176.000000
75%        542.000000
max      95401.000000
Name: stats.owned, dtype: float64
```




```
In [76]: df_data.loc[df_data['stats.owned']==95401]
```

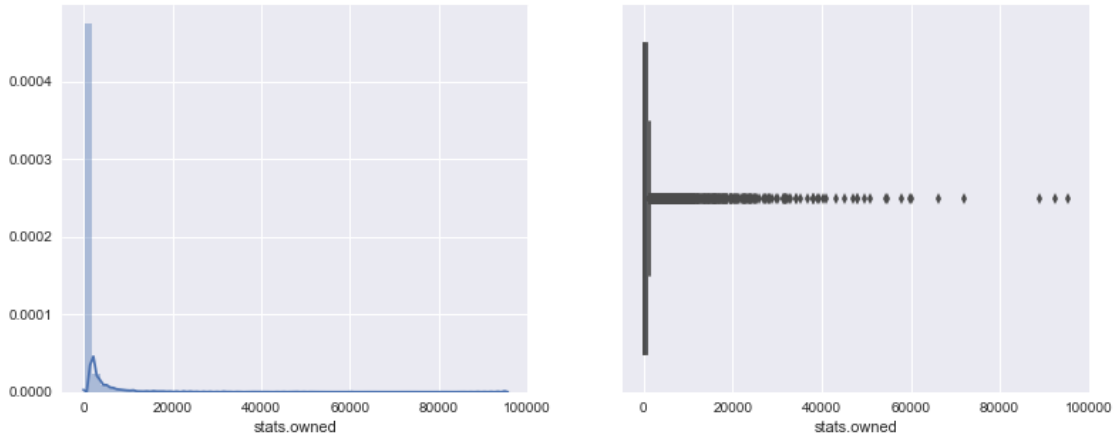
```
Out[76]:    details.name  details.maxplayers  details.minage  details.minplayers  \
         12        Catan                 4.0            10.0                 3.0

             details.playingtime  details.yearpublished  stats.averageweight  \
         12                120.0                 1995.0               2.3603

             stats.average  stats.owned  stats.stddev  Abstract Strategy  \
         12        7.26569      95401.0       1.44842                  0

             Action / Dexterity  Adventure  Age of Reason  American Civil War  \
         12                   0          0              0                   0
```

```
    American Indian Wars  American Revolutionary War  American West  Ancient  \
12                     0                           0              0        0

    Animals  Arabian  Aviation / Flight  Bluffing  Book  Card Game  \
12       0        0                  0         0     0          0

    Children's Game  City Building  Civil War  Civilization  \
12                0              0          0             0

    Collectible Components  Comic Book / Strip  Deduction  Dice  Economic  \
12                      0                   0          0     0         0

    Educational  Electronic  Environmental  Expansion for Base-game  \
12           0           0              0                        0

    Exploration  Fan Expansion  Fantasy  Farming  Fighting  Game System  \
12           0              0        0        0         0            0

    Horror  Humor  Industry / Manufacturing  Korean War  Mafia  Math  \
12       0      0                         0           0      0     0

         ...        Area Control / Area Influence  Area Enclosure  \
12       ...                                    0               0

    Area Movement  Area-Impulse  Auction/Bidding  Betting/Wagering  \
12              0             0                0                 0

    Campaign / Battle Card Driven  Card Drafting  Chit-Pull System  \
12                              0              0                 0

    Co-operative Play  Commodity Speculation  Crayon Rail System  \
12                  0                       0                   0

    Deck / Pool Building  Dice Rolling  Grid Movement  Hand Management  \
12                    0             1              0                1

    Hex-and-Counter  Line Drawing  Memory_mechanics  Modular Board  \
12               0             0                 0              1

    Paper-and-Pencil  Partnerships  Pattern Building  Pattern Recognition  \
12                 0             0                 0                    0

    Pick-up and Deliver  Player Elimination  Point to Point Movement  \
12                    0                   0                        0

    Press Your Luck  Rock-Paper-Scissors  Role Playing  Roll / Spin and Move  \
12               0                    0             0                     0
```

```
    Route/Network Building  Secret Unit Deployment  Set Collection  \
12                       1                       0               0

    Simulation  Simultaneous Action Selection  Singing  Stock Holding  \
12           0                              0        0              0

    Storytelling  Take That  Tile Placement  Time Track  Trading  \
12             0          0               0           0        1

    Trick-taking  Variable Phase Order  Variable Player Powers  Voting  \
12             0                     0                       0       0

    Worker Placement  binned.average  binary.success
12                 0             7.0               1

[1 rows x 147 columns]
```

Without a sufficent enough sample size of games being owned, statistics and data derived from these data may not be robust. Consequently we will limit the minimum number of games that are "owned" to a lower limit of 30.

```
In [77]: df_data6 = df_data5[(df_data5['stats.owned']>=30)]

In [78]: df_data5['stats.owned'].count()-df_data6['stats.owned'].count()

Out[78]: 1199

In [79]: analyze_feature(df_data6, "stats.owned")

count    15262.000000
mean       950.305334
std       3285.209893
min         30.000000
25%         83.000000
50%        202.500000
75%        600.000000
max      95401.000000
Name: stats.owned, dtype: float64
```

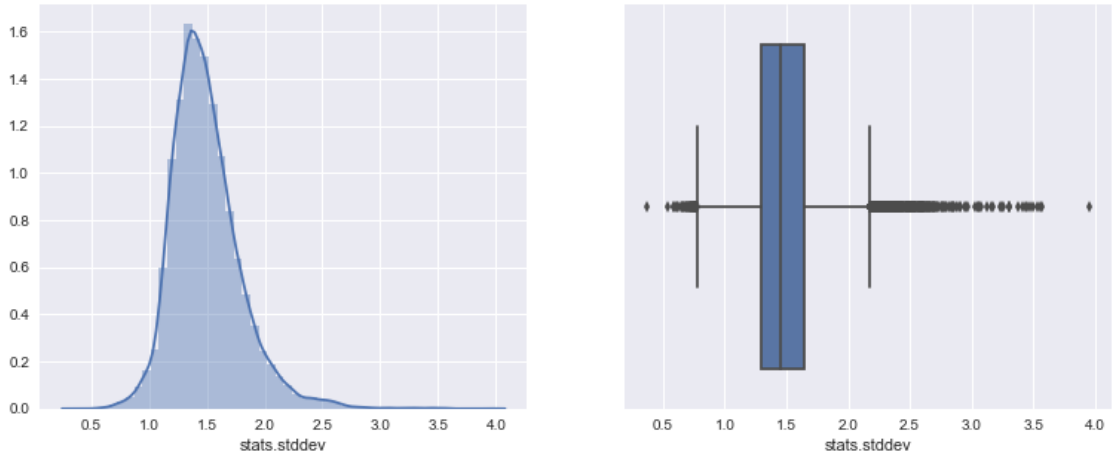


We observe we removed 1199 games because they were owned less than 30 times. Our record set is 15262.

Average Score St.Dev

```
In [80]: analyze_feature(df_data6, "stats.stddev")
```

```
count    15262.000000
mean         1.495717
std          0.301227
min          0.369966
25%          1.297735
50%          1.454630
75%          1.646840
max          3.958430
Name: stats.stddev, dtype: float64
```



Average Score St.dev does not need to be changed too significantly, because there are no errors or supsicious data.

1.5 Creating Data Sets

In order to properly design and validate our learning models, we will need three sets of data:
- Training Data - Development Data - Test Data

First, though, we want to filter our game data by publishing year. As discussed above, we're interested in the shift in board game design in recent years. So, we'll create two data sets: one from 2000 to present, and one from 1980 to 1995.

```
In [81]: new_games_df = df_data6[(df_data6['details.yearpublished'] > 2000)]

         old_games_df = df_data6[(df_data6['details.yearpublished'] > 1980) &
                                 (df_data6['details.yearpublished'] < 1995)]

         print("New games shape: " + str(new_games_df.shape) + "; Old games shape: " + str(old_

New games shape: (10084, 147); Old games shape: (2260, 147)
```

Now we can create the training, dev, and test sets.
We'll use a We'll use a split of 70% / 10% / 20% for the new games, since we'll be developing models based on these games. Each will be randomly selected from the dataframe of entries.
For the old games, we'll separate only a training set and a test set. (We won't do model development with the old games; we'll simply apply the model developed for the newer games.)

```
In [82]: #Split the new/old game DF into features and outcomes:
         new_games_df_features = new_games_df.drop(["stats.average", "binned.average",
                                                    "stats.stddev", "stats.owned", "details.nam
         new_game_df_outcomes = new_games_df[["stats.average", "binned.average", "stats.stddev"
         old_games_df_features = old_games_df.drop(["stats.average", "binned.average",
                                                    "stats.stddev", "stats.owned", "details.nam
         old_games_df_outcomes = old_games_df[["stats.average", "binned.average", "stats.stddev

In [83]: # create a random arrangement of the data sets
         # New Games
         np.random.seed(0)
         shuffle_new = np.random.permutation(np.arange(new_games_df_features.shape[0]))
         shuffle_old = np.random.permutation(np.arange(old_games_df_features.shape[0]))
         #Convert all data frames into Nump Arrays:
         new_games_df_feat = new_games_df_features.values
         new_games_df_out = new_game_df_outcomes.values
         old_game_df_feat = old_games_df_features.values
         old_game_df_out = old_games_df_outcomes.values

In [84]: # create a random arrangement of the data sets

         shuffled_new_feat = new_games_df_feat[shuffle_new]
```

```
shuffled_new_out = new_games_df_out[shuffle_new]

shuffled_old_feat = old_game_df_feat[shuffle_old]
shuffled_old_out = old_game_df_out[shuffle_old]

print(shuffled_new_feat.shape) #we expect to still see 10084 rows and 141 features
print(shuffled_new_out.shape) #we expect to still see 10084 rows and 5 features
print(shuffled_old_feat.shape) #we expect to still see 2260 rows and 141 features
print(shuffled_old_out.shape) #we expect to still see 2260 rows and 5 features
```

```
(10084, 141)
(10084, 5)
(2260, 141)
(2260, 5)
```

In [85]:
```
#Splitting into Training, Dev, and Test Sets - Note: won't include a dev set for Old

#Get the number of Records
n_new_games = shuffled_new_feat.shape[0]
n_old_games = shuffled_old_feat.shape[0]
print(n_new_games)
print(n_old_games)

#Split:
```

```
10084
2260
```

In [86]:
```
#New Game data sets

new_split1 = int(np.floor(.7*n_new_games)) # 70% for training
new_split2 = new_split1 + int(np.floor(.1*n_new_games)) + 1  # 10% for dev

print(new_split1, new_split2)

new_train_data = shuffled_new_feat[:new_split1]
new_train_labels = shuffled_new_out[:new_split1]

new_dev_data = shuffled_new_feat[new_split1:new_split2]
new_dev_labels = shuffled_new_out[new_split1:new_split2]

new_test_data = shuffled_new_feat[new_split2:]
new_test_labels = shuffled_new_out[new_split2:]

print("Training shape: " + str(new_train_data.shape) + "; Dev shape: " + str(new_dev_
print(np.ma.size(new_train_data,0)+np.ma.size(new_dev_data,0)+np.ma.size(new_test_data
```

```
7058 8067
Training shape: (7058, 141); Dev shape: (1009, 141); Test shape: (2017, 141)
10084
```

```
In [87]: # Old game data sets
         old_split1 = int(np.floor(.7*n_old_games)) # 70% for training


         old_train_data = shuffled_old_feat[:old_split1]
         old_train_labels = shuffled_old_out[:old_split1]

         old_test_data = shuffled_old_feat[old_split1:]
         old_test_labels = shuffled_old_out[old_split1:]

         print("Training shape: " + str(old_train_data.shape) + "; Test shape: " + str(old_tes
         print(np.ma.size(old_train_data,0)+np.ma.size(old_test_data,0))
```

```
Training shape: (1582, 141); Test shape: (678, 141)
2260
```

1.6 Modeling - kNN

The first model we are attempting is a k-Nearest-Neighbors model. This will find board games similar to the one in question, and use them to predict the rating of the unknown game. This seems similar to how real life works - players use similar games to decide whether to buy a new game - so it's worth a shot here!

```
In [88]: from sklearn.neighbors import KNeighborsClassifier


         k_values = [1,3,5,7,9]

         for k in k_values:
             ## MODEL USING BINARY SUCCESS (TWO CATEGORIES FOR OUTCOME)
             binary_kNN = KNeighborsClassifier(n_neighbors = k)
             binary_kNN.fit(new_train_data, new_train_labels[:,4]) # binary_success is in colu
             # take a stab at the dev set
             y_pred = binary_kNN.predict(new_dev_data)
             # accuracy
             binary_acc = round(sum(y_pred == new_dev_labels[:,4])/new_dev_labels.shape[0],3)*
             print("\nThe binary classifier with k=" + str(k) + " had an accuracy of " + str(bi

             ## MODEL USING STAR RATING, 0-10 (TEN CATEGORIES FOR OUTCOME)
             ten_kNN = KNeighborsClassifier(n_neighbors = k)
             ten_kNN.fit(new_train_data, new_train_labels[:,1]) # binned star rating is in col
             # take a stab at the dev set
             y_pred = ten_kNN.predict(new_dev_data)
```

```
            # accuracy
            ten_acc = round(sum(y_pred == new_dev_labels[:,1])/new_dev_labels.shape[0],3)*100
            # what if you consider +/- 1 star to be still "accurate"?
            ten_acc2 = round((
                sum(y_pred == new_dev_labels[:,1]) +
                sum(y_pred == new_dev_labels[:,1]-1) + sum(y_pred == new_dev_labels[:,1]+1)
                )/new_dev_labels.shape[0],3)*100
            print("The 1-10 scale classifier with k=" + str(k) + " had an accuracy of "
                  + str(ten_acc) + "% (and a +/-1 star accuracy of " + str(ten_acc2) + "%)." )

        # print the binary values the model gets wrong
        print("\nThe ratings misclassified by the binary classifier were: ")
        y_pred = binary_kNN.predict(new_dev_data)
        binary_errors = new_dev_labels[y_pred != new_dev_labels[:,4],:]
        print(str(binary_errors[0:15,1]) + "-- actual values")

        # print the binned values the model gets wrong
        print("\nThe ratings misclassified by the binned 0-10 classifier were: ")
        y_pred = ten_kNN.predict(new_dev_data)
        ten_errors = new_dev_labels[y_pred != new_dev_labels[:,1],:]
        print(str(ten_errors[0:15,1]) + "-- actual values")
        print(str(y_pred[y_pred != new_dev_labels[:,1]][0:15]) + "-- predicted values")
```

```
The binary classifier with k=1 had an accuracy of 77.3%.
The 1-10 scale classifier with k=1 had an accuracy of 39.2% (and a +/-1 star accuracy of 88.8%)

The binary classifier with k=3 had an accuracy of 79.2%.
The 1-10 scale classifier with k=3 had an accuracy of 39.3% (and a +/-1 star accuracy of 87.8%)

The binary classifier with k=5 had an accuracy of 79.2%.
The 1-10 scale classifier with k=5 had an accuracy of 42.9% (and a +/-1 star accuracy of 92.6%)

The binary classifier with k=7 had an accuracy of 79.4%.
The 1-10 scale classifier with k=7 had an accuracy of 44.1% (and a +/-1 star accuracy of 93.0%)

The binary classifier with k=9 had an accuracy of 78.9%.
The 1-10 scale classifier with k=9 had an accuracy of 44.4% (and a +/-1 star accuracy of 92.9%)

The ratings misclassified by the binary classifier were:
[ 6.  7.  7.  6.  6.  7.  6.  7.  6.  7.  6.  7.  6.  7.  7.]-- actual values

The ratings misclassified by the binned 0-10 classifier were:
[ 6.  5.  7.  6.  7.  5.  6.  5.  8.  6.  7.  8.  6.  5.  6.]-- actual values
[ 5.  4.  6.  7.  6.  6.  7.  6.  7.  5.  6.  7.  5.  6.  7.]-- predicted values
```

The kNN model works reasonably well! We are able to achieve about 80% accuracy categorizing games into binary 'success' (rating > 7 stars) and 'failure' (rating < 7 stars) categories. We can

see from the printed error values that most of the errors occur in the 6-7 star range, which is right on the border of "success".
If we maintain the full spread of the 10 binned star-rating categories (integer values 1,2,3...10), we are able to achieve only about 45% accuracy. However, if we concede that plus-or-minus one star is still an "accurate" prediction of rating, we are able to get an impressive accuracy above 90%! The output immediately above shows the true star rating listed above the star values predicted by the model. We can see that again, the mid-scale (5-8 star) range is the issue.

Based on our trials with the development data, we can see that the parameter k=5 creates the best kNN model. We'll save the k=5 models for evaluation on our test data later.

```
In [89]: # Keep k=5 models
         binary_kNN = KNeighborsClassifier(n_neighbors = k)
         binary_kNN.fit(new_train_data, new_train_labels[:,4]) # binary_success is in column 4

         ten_kNN = KNeighborsClassifier(n_neighbors = k)
         ten_kNN.fit(new_train_data, new_train_labels[:,1]) # binned star rating is in column

         print("kNN models saved!")
```

```
kNN models saved!
```

1.7 Modeling - Naive Bayes

Next, we'll attempt a Naive Bayes model to classify the games. Since our features are multi-valued, we'll use the MultinomialNB option. Naive Bayes assumes that all features are conditionally independent and predicts the probability of each label given the feature values. We know that at least some of our features are not independent (for example, gameplay mechanics likely have some effect on playing time), so the naive assumption is certainly not entirely accurate. However, we'll explore this model anyway.

```
In [90]: from sklearn.naive_bayes import MultinomialNB

         ## BINARY SUCCESS MODEL
         # Create and train the model
         binary_NBmodel = MultinomialNB()
         binary_NBmodel.fit(new_train_data, new_train_labels[:,4]) # binary success, column 4

         # Predict values for dev data
         y_pred = binary_NBmodel.predict(new_dev_data)
         acc = round(sum(y_pred == new_dev_labels[:,4])/len(new_dev_labels), 3)*100
         print("The Naive Bayes classifier predicted binary success with " + str(acc) + "% accu

         ## BINNED SCALE MODEL
         # Create and train the model
         ten_NBmodel = MultinomialNB()
         ten_NBmodel.fit(new_train_data, new_train_labels[:,1]) # binned 0-10 success, column

         # Predict values for dev data
```

```
        y_pred = ten_NBmodel.predict(new_dev_data)
        acc = round(sum(y_pred == new_dev_labels[:,1])/len(new_dev_labels), 3)*100
        # what if you consider +/- 1 star to be still "accurate"?
        acc2 = round((
            sum(y_pred == new_dev_labels[:,1]) +
            sum(y_pred == new_dev_labels[:,1]-1) + sum(y_pred == new_dev_labels[:,1]+1)
            )/new_dev_labels.shape[0],3)*100
        print("and predicted the 1-10 binned star rating with " + str(acc) +
              "% accuracy (+/-1 star with " + str(acc2) + "%). \n")
```

```
The Naive Bayes classifier predicted binary success with 73.6% accuracy
and predicted the 1-10 binned star rating with 34.1% accuracy (+/-1 star with 84.2%).
```

Well, Naive Bayes didn't achieve accuracy as high as kNN. Still, there is one benefit - Naive Bayes is a generative model, so we could actually use this model to "create" a new board game. The model can tell us the ideal playtime, number of players, mechanic, etc. Then it would be up to us to design a story and title to fit that mold!
In that spirit, let's take a look at which game mechanics and themes are the most likely to create a successful game.

```
In [91]: # feature probabilities for SUCCESS (outcome category 1)
         probs = np.exp(binary_NBmodel.feature_log_prob_[1,:])
         coefs = binary_NBmodel.coef_

         # game themes are in columns 6 through 89
         new_games_df_features.columns.tolist()[6:90]
         top_feat_inds = np.fliplr([np.argsort(probs[6:90])[-5:]])[0]

         for i in top_feat_inds:
             print("The theme " + new_games_df_features.columns.tolist()[6:90][i] +
                   " has a probability of " +  str(round(probs[6:90][i],5)*100) + "% in a succ

         print("\n")

         # game mechanics are in columns 91 through the end
         new_games_df_features.columns.tolist()[90:]
         top_feat_inds = np.fliplr([np.argsort(probs[90:])[-5:]])[0]

         for i in top_feat_inds:
             print("The mechanic " + new_games_df_features.columns.tolist()[90:][i] +
                   " has a probability of " +  str(round(probs[90:][i],5)*100) + "% in a succes
```

```
The theme Wargame has a probability of 0.012% in a successful game.
The theme Card Game has a probability of 0.011% in a successful game.
The theme Fantasy has a probability of 0.008% in a successful game.
The theme Economic has a probability of 0.006% in a successful game.
The theme Science Fiction has a probability of 0.005% in a successful game.
```

```
The mechanic Dice Rolling has a probability of 0.015% in a successful game.
The mechanic Hand Management has a probability of 0.012% in a successful game.
The mechanic Variable Player Powers has a probability of 0.01% in a successful game.
The mechanic Area Control / Area Influence has a probability of 0.007% in a successful game.
The mechanic Card Drafting has a probability of 0.006% in a successful game.
```

Looks like the more successful modern games have themes of wargames, card games, fantasy, economy, and science fiction. That certainly matches what we know from experience! The preferred mechanics are dice rolling, hand management, variable player powers, area control, and card drafting (aka deck building). Again, that is what we have seen in our favorite games!

1.8 Modeling - Linear Regression

Finally, we'll try a linear regression model for board game rating. This time, we'll use the full continous scale of rating (1-10, with decimal values included), since the model doesn't have to assign outcomes to countable categories.

```
In [92]: from sklearn.linear_model import LinearRegression

         # linear regression model
         Regmodel = LinearRegression()
         Regmodel.fit(new_train_data, new_train_labels[:,0]) # raw star values are in column 0

         print("The linear regression model achieves an R2 value of " + str(Regmodel.score(new

         y_pred = Regmodel.predict(new_dev_data)
         print("At worst, the predicted score was off by " + str(round(np.max(y_pred - new_dev
```

```
The linear regression model achieves an R2 value of 0.460947761954
At worst, the predicted score was off by 3.058 stars.
```

This linear regression model achieved an R2 of less than 0.5, and star ratings predicted within about 3 stars of the true value. Not the best performance - our kNN model is still winning. However, linear regression is wonderfully interpretable - the model here tells us the most important features in predicting success of a game.

```
In [93]: # what are the most important features?
         coefs = Regmodel.coef_
         top_feat_inds = np.fliplr([np.argsort(coefs)[-7:]])[0]
         bottom_feat_inds = np.argsort(coefs)[0:7]

         # features that bring rating up
         for i in top_feat_inds:
             print("The feature " + new_games_df_features.columns.tolist()[i] +
```

```
                " has a coefficient of " +  str(coefs[i]))
        print("\n")
        # features that bring rating down
        for i in bottom_feat_inds:
            print("The feature " + new_games_df_features.columns.tolist()[i] +
                " has a coefficient of " +  str(coefs[i]))
```

```
The feature Vietnam War has a coefficient of 0.491231458653
The feature American Indian Wars has a coefficient of 0.465557998552
The feature Napoleonic has a coefficient of 0.462687194417
The feature Crayon Rail System has a coefficient of 0.456555073874
The feature Game System has a coefficient of 0.427079240971
The feature Book has a coefficient of 0.412035099255
The feature stats.averageweight has a coefficient of 0.410222939636


The feature Roll / Spin and Move has a coefficient of -0.331455259955
The feature Movies / TV / Radio theme has a coefficient of -0.285456798849
The feature Mature / Adult has a coefficient of -0.284711811953
The feature Singing has a coefficient of -0.183744475696
The feature Number has a coefficient of -0.181778425733
The feature Zombies has a coefficient of -0.162301453659
The feature Trading has a coefficient of -0.146409417658
```

Looks like having a game themed on a historical war (Vietnam, American Indian, or Napoleonic) brings success, as does having a book theme. The "stats.averageweight" feature corresponds to gameplay difficulty; seems like more complex games help increase the player satisfaction rating!
On the other hand, having a game that requires rolling/spinning and moving (like traditional Monopoly or Life), singing, or trading all decrease satisfaction. Having mature/adult content, a movie/TV/radio theme, or a zombie theme also spells doom for game reception.

2 Evaluation of Models on Test Data

This section passes the test data through the three model types we built above. As we saw with the dev data, the kNN model performs the best.

In [94]:
```
## kNN MODELS
    # BINARY - evaluate the test data
y_pred = binary_kNN.predict(new_test_data)
    # accuracy
acc = round(sum(y_pred == new_test_labels[:,4])/new_test_labels.shape[0],3)*100
print("\nThe binary kNN classifier had an accuracy of " + str(acc) + "%." )

    # BINNED - evaluate the test data
y_pred = ten_kNN.predict(new_test_data)
    # accuracy
```

```
        acc = round(sum(y_pred == new_test_labels[:,1])/new_test_labels.shape[0],3)*100
            # what if you consider +/- 1 star to be still "accurate"?
        acc2 = round((
            sum(y_pred == new_test_labels[:,1]) +
            sum(y_pred == new_test_labels[:,1]-1) + sum(y_pred == new_test_labels[:,1]+1)
            )/new_test_labels.shape[0],3)*100
        print("The binned 1-10 scale kNN classifier had an accuracy of "
              + str(acc) + "%, or " + str(acc2) +  "% if you allow for +/-1 star.")
```

```
The binary kNN classifier had an accuracy of 80.6%.
The binned 1-10 scale kNN classifier had an accuracy of 48.4%, or 93.5% if you allow for +/-1 s
```

```
In [95]: ## Naive Bayes MODELS
            # BINARY - evaluate the test data
        y_pred = binary_NBmodel.predict(new_test_data)
            # accuracy
        acc = round(sum(y_pred == new_test_labels[:,4])/new_test_labels.shape[0],3)*100
        print("\nThe binary Naive Bayes classifier had an accuracy of " + str(acc) + "%." )

            # BINNED - evaluate the test data
        y_pred = ten_NBmodel.predict(new_test_data)
            # accuracy
        acc = round(sum(y_pred == new_test_labels[:,1])/new_test_labels.shape[0],3)*100
            # what if you consider +/- 1 star to be still "accurate"?
        acc2 = round((
            sum(y_pred == new_test_labels[:,1]) +
            sum(y_pred == new_test_labels[:,1]-1) + sum(y_pred == new_test_labels[:,1]+1)
            )/new_test_labels.shape[0],3)*100
        print("The binned 1-10 scale Naive Bayes classifier had an accuracy of "
              + str(acc) + "%, or " + str(acc2) +  "% if you allow for +/-1 star.")
```

```
The binary Naive Bayes classifier had an accuracy of 76.4%.
The binned 1-10 scale Naive Bayes classifier had an accuracy of 35.1%, or 84.9% if you allow fo
```

```
In [96]: ## Linear Regression model
        print("The linear regression model achieves an R2 value of " + str(Regmodel.score(new_
        y_pred = Regmodel.predict(new_test_data)
        print("On average, the predicted score was off by " + str(round(np.average(np.abs(y_pr
        print("At worst, the predicted score was off by " + str(round(np.max(y_pred - new_test
```

```
The linear regression model achieves an R2 value of 0.437876935045
On average, the predicted score was off by 0.509 stars.
At worst, the predicted score was off by 4.966 stars.
```

3 Analysis of "Old" Games

Now that we understand what features make a modern board game successful, we can analyze games from 1980 through 1995 to see how things have changed. We'll use the Naive Bayes approach to explore the most important features, since it had a better accuracy than Linear Regression in our experiments.

In [97]:
```
## BINARY SUCCESS MODEL
# Create and train the model
binary_NBmodel = MultinomialNB()
binary_NBmodel.fit(old_train_data, old_train_labels[:,4]) # binary success, column 4

# Predict values for dev data
y_pred = binary_NBmodel.predict(old_test_data)
acc = round(sum(y_pred == old_test_labels[:,4])/len(old_test_labels), 3)*100
print("The Naive Bayes classifier predicted binary success with " + str(acc) + "% accu

## BINNED SCALE MODEL
# Create and train the model
ten_NBmodel = MultinomialNB()
ten_NBmodel.fit(old_train_data, old_train_labels[:,1]) # binned 0-10 success, column

# Predict values for dev data
y_pred = ten_NBmodel.predict(old_test_data)
acc = round(sum(y_pred == old_test_labels[:,1])/len(new_dev_labels), 3)*100
# what if you consider +/- 1 star to be still "accurate"?
acc2 = round((
    sum(y_pred == old_test_labels[:,1]) +
    sum(y_pred == old_test_labels[:,1]-1) + sum(y_pred == old_test_labels[:,1]+1)
    )/old_test_labels.shape[0],3)*100
print("and predicted the 1-10 binned star rating with " + str(acc) +
      "% accuracy (+/-1 star with " + str(acc2) + "%). \n")
```

```
The Naive Bayes classifier predicted binary success with 78.0% accuracy
and predicted the 1-10 binned star rating with 17.2% accuracy (+/-1 star with 75.1%).
```

In [98]:
```
# feature probabilities for SUCCESS (outcome category 1)
probs = np.exp(binary_NBmodel.feature_log_prob_[1,:])
coefs = binary_NBmodel.coef_

# game themes are in columns 6 through 89
old_games_df_features.columns.tolist()[6:90]
top_feat_inds = np.fliplr([np.argsort(probs[6:90])[-5:]])[0]

for i in top_feat_inds:
    print("The theme " + old_games_df_features.columns.tolist()[6:90][i] +
          " has a probability of " +  str(round(probs[6:90][i],5)*100) + "% in a succe
```

```
print("\n")

# game mechanics are in columns 91 through the end
old_games_df_features.columns.tolist()[90:]
top_feat_inds = np.fliplr([np.argsort(probs[90:])[-5:]])[0]

for i in top_feat_inds:
    print("The mechanic " + old_games_df_features.columns.tolist()[90:][i] +
          " has a probability of " +  str(round(probs[90:][i],5)*100) + "% in a succes
```

```
The theme Wargame has a probability of 0.031% in a successful game.
The theme World War II has a probability of 0.014% in a successful game.
The theme Science Fiction has a probability of 0.005% in a successful game.
The theme Miniatures has a probability of 0.005% in a successful game.
The theme Modern Warfare has a probability of 0.004% in a successful game.
```

```
The mechanic Hex-and-Counter has a probability of 0.022% in a successful game.
The mechanic Dice Rolling has a probability of 0.015% in a successful game.
The mechanic Simulation has a probability of 0.011% in a successful game.
The mechanic Variable Player Powers has a probability of 0.004% in a successful game.
The mechanic Action Point Allowance System has a probability of 0.003% in a successful game.
```

Looks like things have in fact changed a bit in the last 25 years! Wargames and Science Fiction were popular back then (and still are). However, the other front-running themes were World War II, Mineatures, and Modern Warfare. Perhaps that makes sense, considering that 1980 was still feeling the impact of American involvement in World War II and Vietnam.
As for mechanics, dice rolling and variable player powers were popular - and still are today. Otherwise, hex-and-counter, simulation, and action point allowance games were the trends of the last millenium.

4 Summary

In this analysis, we constructed three types of models to explore the features that make modern board games successful. * A k-Nearest-Neighbors approach gave us the best results, with the ability to predict the 0-10 star rating with over 90% accuracy (allowing +/-1 star margin) * Naive Bayes modeling helped us understand the themes and gameplay mechanics that most correspond to high ratings * Linear Regression models explored the full, continuous rating scale and helped us understand which features most strongly predict success - and failure!

We learned that dice-rolling and variable player power games are perennial favorites, as are games with Science Fiction and Wargame themes. However, modern games have pivoted some, and new mechanics such as hand management, card drafting, and area control have replaced the old world of miniatures and simulation.