assignment 10 (old)

March 8, 2022

1 I ended up not using this data set in the end, as the correlations were very weak

```
[1]: import pandas as pd
```

```
[2]: import seaborn as sns
```

```
[3]: df = pd.read_csv('steam.csv', sep=',')
```

```
[4]: df.pop('appid')
     # Convert english too boolean
     df['english'] = df['english'].astype('bool')
     # set release date to datetime
     df['release_date'] = pd.to_datetime(df['release_date'])
     # create 3 seperate platform fields instead of 1
     df['windows'], df['mac'], df['linux'] = df['platforms'].apply(lambda x:␣
      ↪'windows' in x),df['platforms'].apply(lambda x: 'mac' in x),df['platforms'].
      ↪apply(lambda x: 'linux' in x)
     df.drop(columns='platforms', inplace=True)
     df['owners_low'] = df['owners'].apply(lambda x: x.split('-')[0]).astype('int')
     df['owners_high'] = df['owners'].apply(lambda x: x.split('-')[1]).astype('int')
     df['release_year'] = df['release_date'].dt.year
     df.drop(columns='owners', inplace=True)
     genres = df['genres'].apply(lambda x: x.split(';')[0])
```

```
[5]: medianPlaytimeFilter = df['median_playtime']> 0.5
     ownersFilter = df['owners_low'] > 20000 #lowest range above 0
     reviewFilter = df['positive_ratings'] > 5
     noFreeGameFilter = df['price'] > 0.1
```

```
[7]: gameCorrelations = df.corr()
     gameCorrelations.style.background_gradient(cmap='coolwarm', axis=None).
      ↪set_precision(2)
```

```
C:\Users\Stijn\AppData\Local\Temp/ipykernel_11924/4049258250.py:2:
FutureWarning: this method is deprecated in favour of
`Styler.format(precision=..)`
```

```
  gameCorrelations.style.background_gradient(cmap='coolwarm',
axis=None).set_precision(2)
```

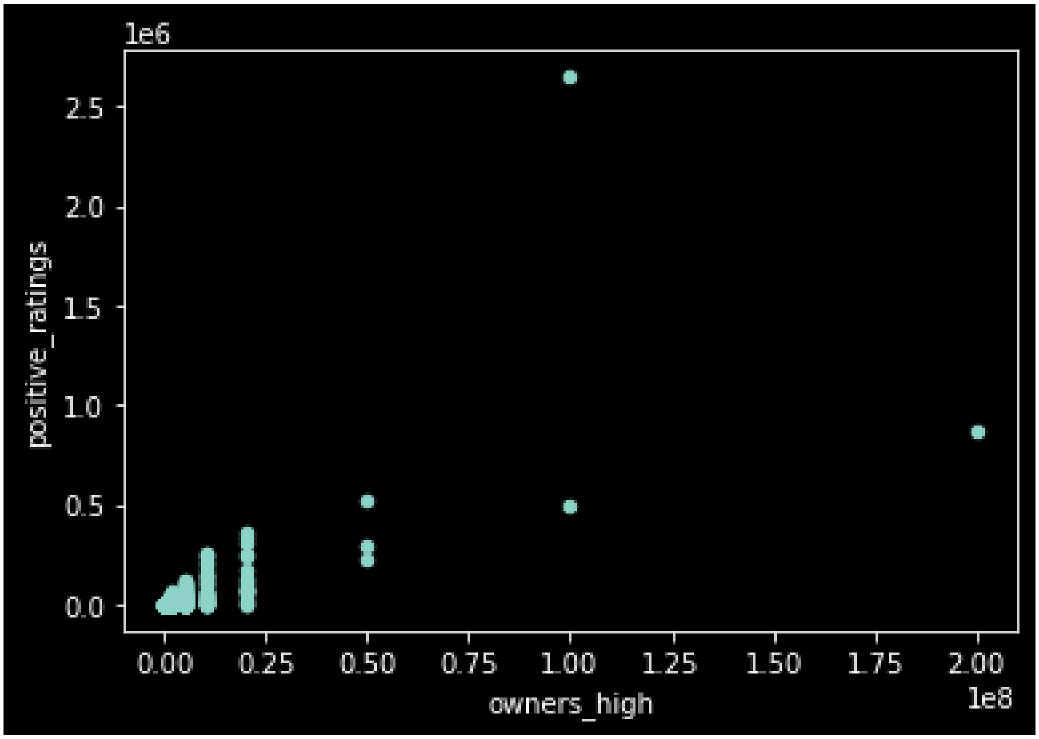
```
[7]: <pandas.io.formats.style.Styler at 0x29842e41100>
```

```
[14]: correlation2 = df[['owners_high','positive_ratings']]
      correlation2.corr()
```

```
[14]:                   owners_high  positive_ratings
      owners_high          1.000000          0.712447
      positive_ratings     0.712447          1.000000
```

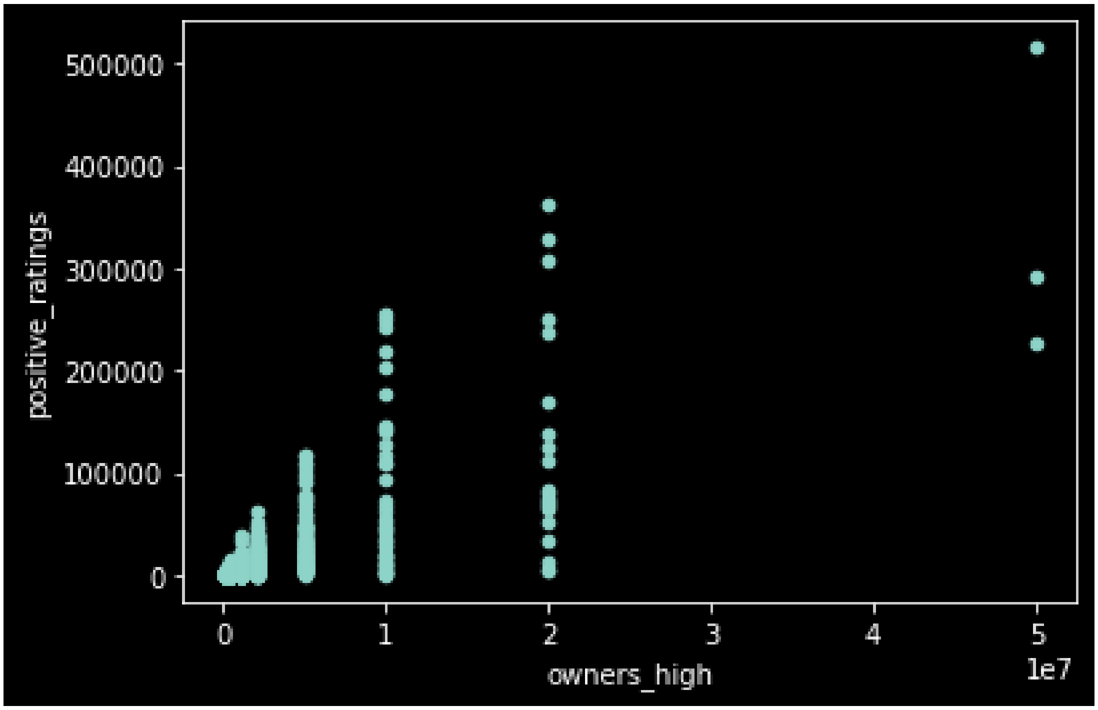
```
[11]: df.plot(kind='scatter', x='owners_high',y='positive_ratings')
```

```
[11]: <AxesSubplot:xlabel='owners_high', ylabel='positive_ratings'>
```



```
[10]: df[df['owners_high'] < 100000000].plot(kind='scatter',␣
       ↪x='owners_high',y='positive_ratings')
```

```
[10]: <AxesSubplot:xlabel='owners_high', ylabel='positive_ratings'>
```



It does match my expectation, as you will have more (positive) reviews if more people play your game