Analyzing the Reception of PC Video Games through Steam Data

By Afnan Dzaharudin

Introduction

I like video games

I like video games a lot

Owned Products 195 products

Show 25 entries. Hold Shift to sort multiple columns. Search:

Name	Price / Hour	Price	Time	Rating
Team Fortress 2	-	Free	1,022.5h	93.02%
Terraria	0.01	$9.99	1,020.4h	97.17%
Dota 2	-	Free	975.8h	83.33%
Warframe	-	Free	912.2h	89.66%
Unturned	-	Free	300.9h	90.62%
PAYDAY 2	0.06	$9.99	162.7h	88.14%
Starbound	0.09	$14.99	158.6h	91.77%
Stardew Valley	0.10	$14.99	142.8h	97.05%
SMITE	-	Free	134.2h	79.54%
Apex Legends	-	Free	125.0h	85.91%
The Elder Scrolls Online	0.06	$5.99	96.3h	80.97%
Castle Crashers	0.16	$14.99	94.7h	94.70%
Space Engineers	0.21	$19.99	93.4h	87.21%
For The King	0.08	$6.79	90.4h	87.09%
Risk of Rain 2	0.29	$24.99	86.2h	95.27%
Monster Hunter: World	0.25	$19.79	78.1h	84.02%

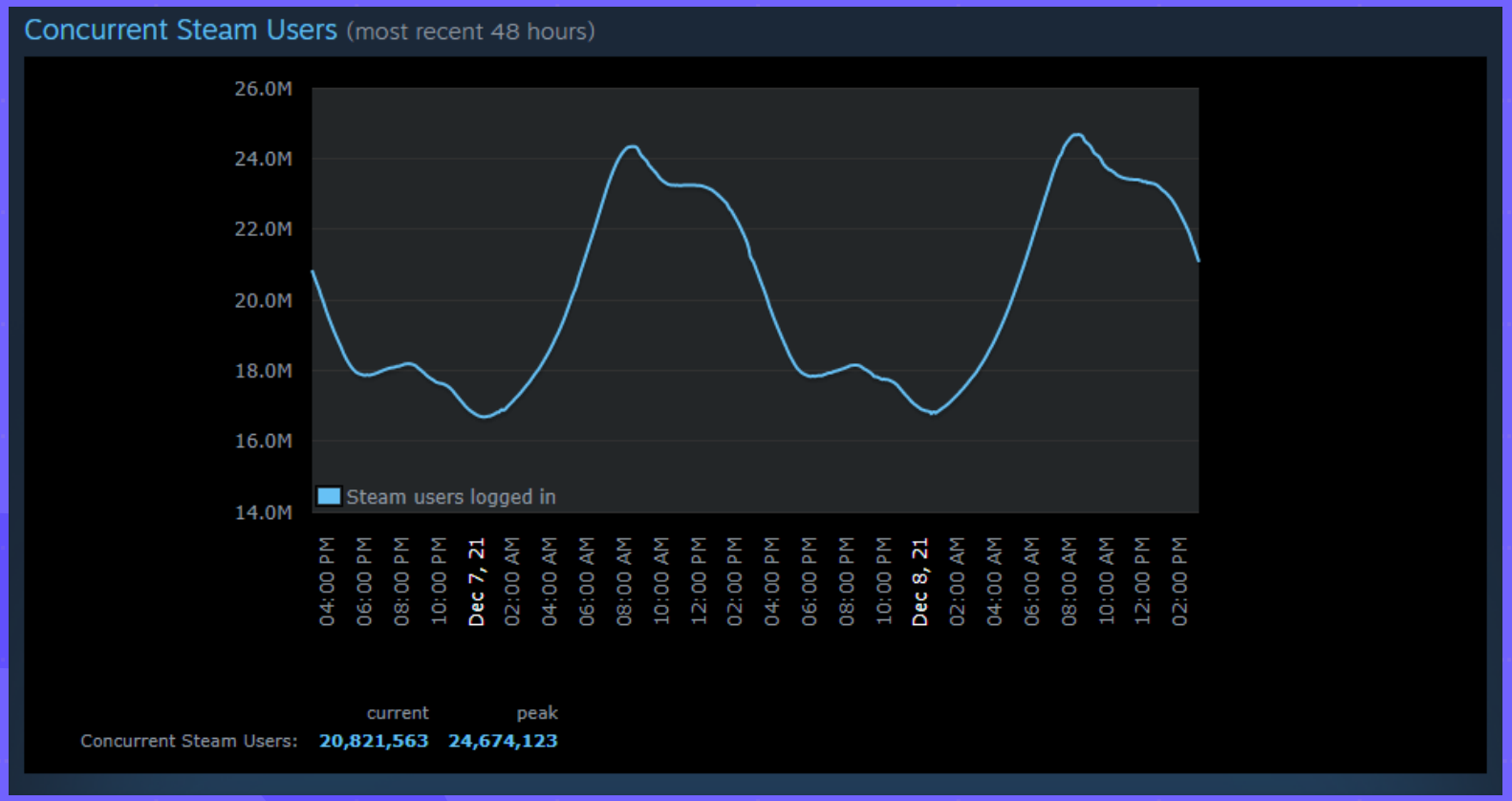
Dataset

	appid	name	release_date	developer	publisher	positive_reviews	negative_reviews	owners	average_forever	average_2weeks	median_forever	median_2weeks	price
166	271590	Grand Theft Auto V	2015-04-13	Rockstar North	Rockstar Games	1046867	200830	20,000,000 - 50,000,000	12414	685	6283	187	29.98
165	346110	ARK: Survival Evolved	2017-08-27	Studio Wildcard, Instinct Games, Efecto Studio...	Studio Wildcard	388725	91242	10,000,000 - 20,000,000	13700	1119	1575	433	29.99

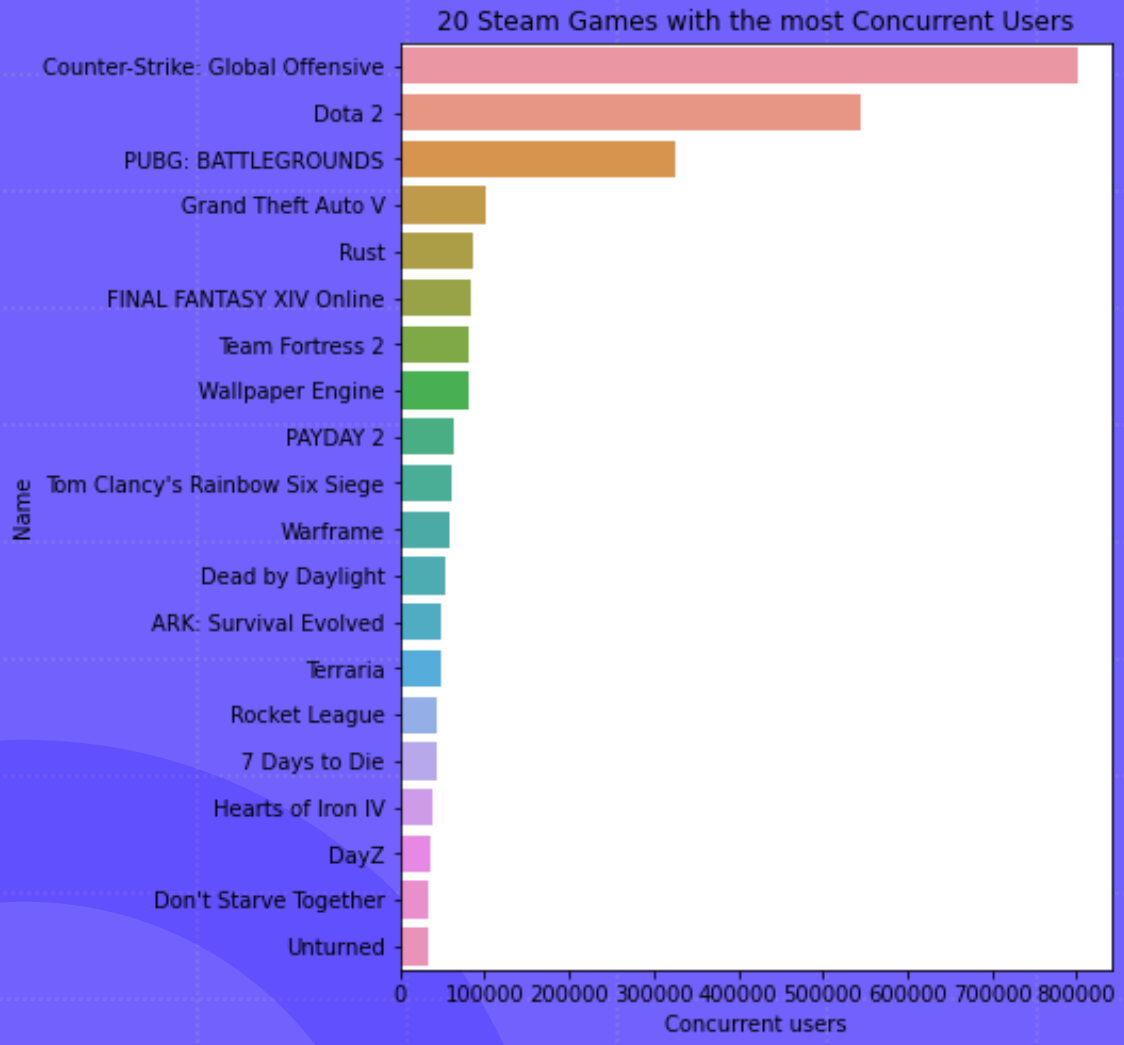
ccu	english	categories	genres	is_multiplayer	is_indie	release_year
101552	1	Single-player;Multi-player;Steam Achievements;...	Action;Adventure	True	False	2015
49997	1	Single-player;Multi-player;MMO;Co-op;Steam Ach...	Action;Adventure;Indie;Massively Multiplayer;RPG	True	True	2017

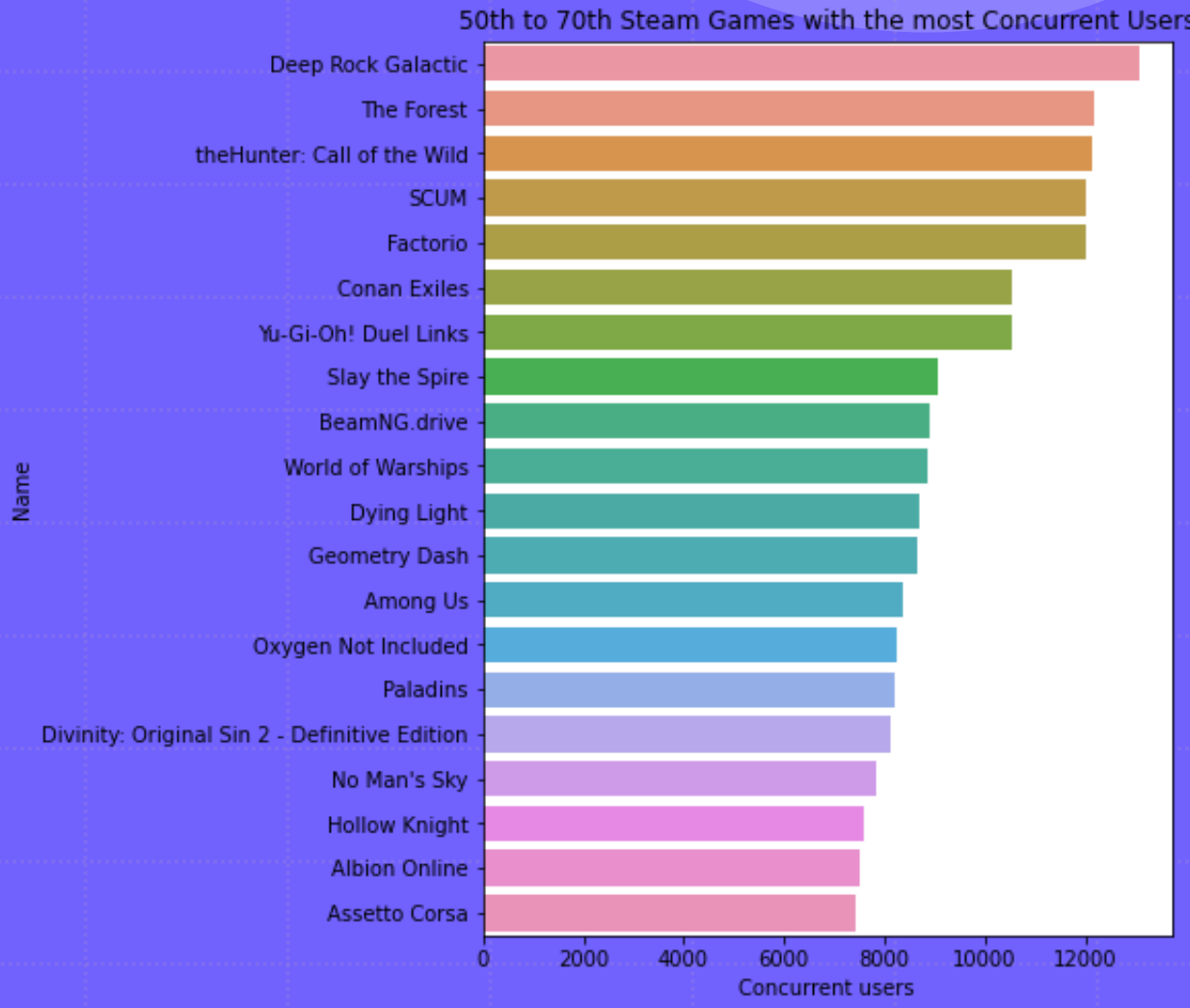
Descriptive Analytics

CCU (Concurrent Users)



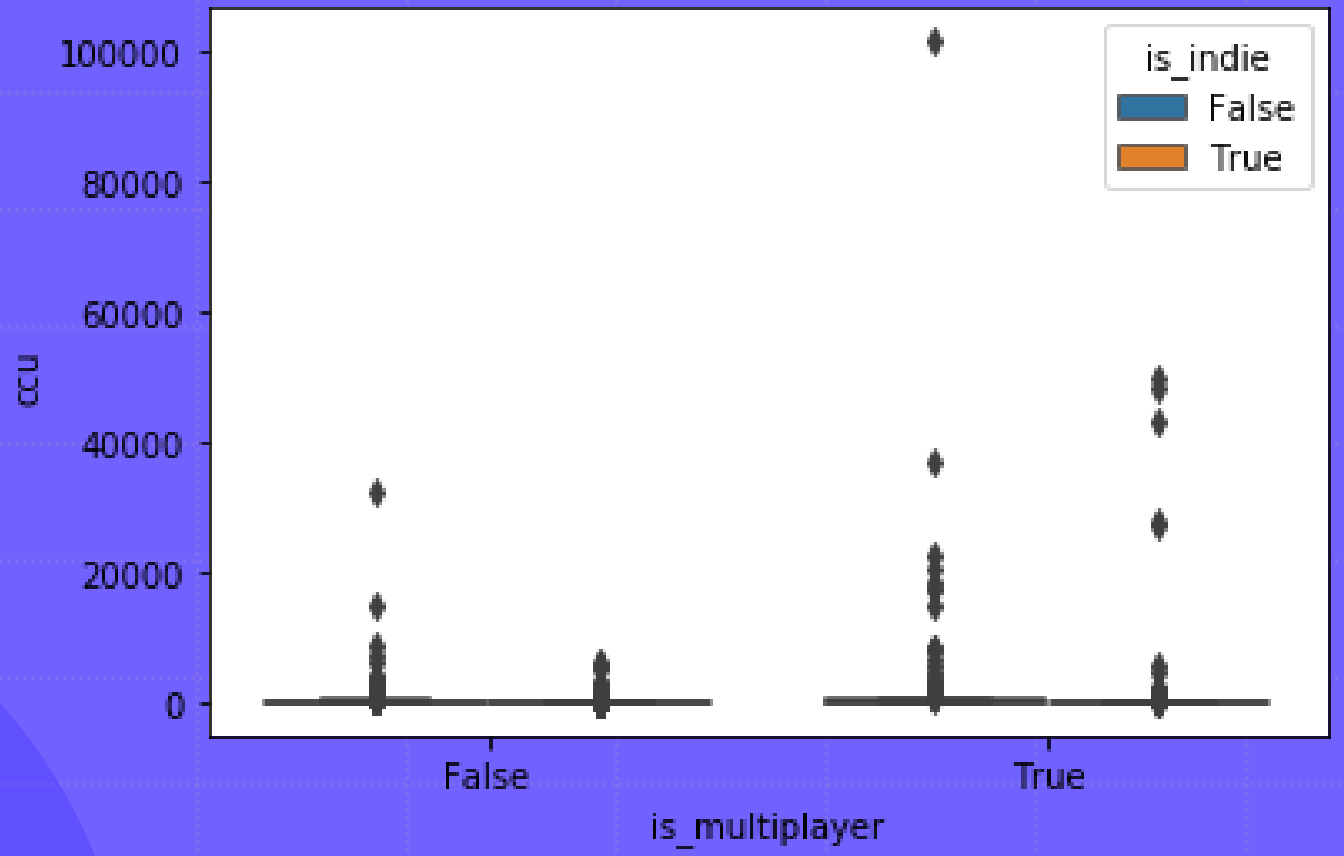
Descriptive Analytics





Descriptive Analytics

Concurrent Users ~ is_multiplayer + is_indie



Descriptive Analytics

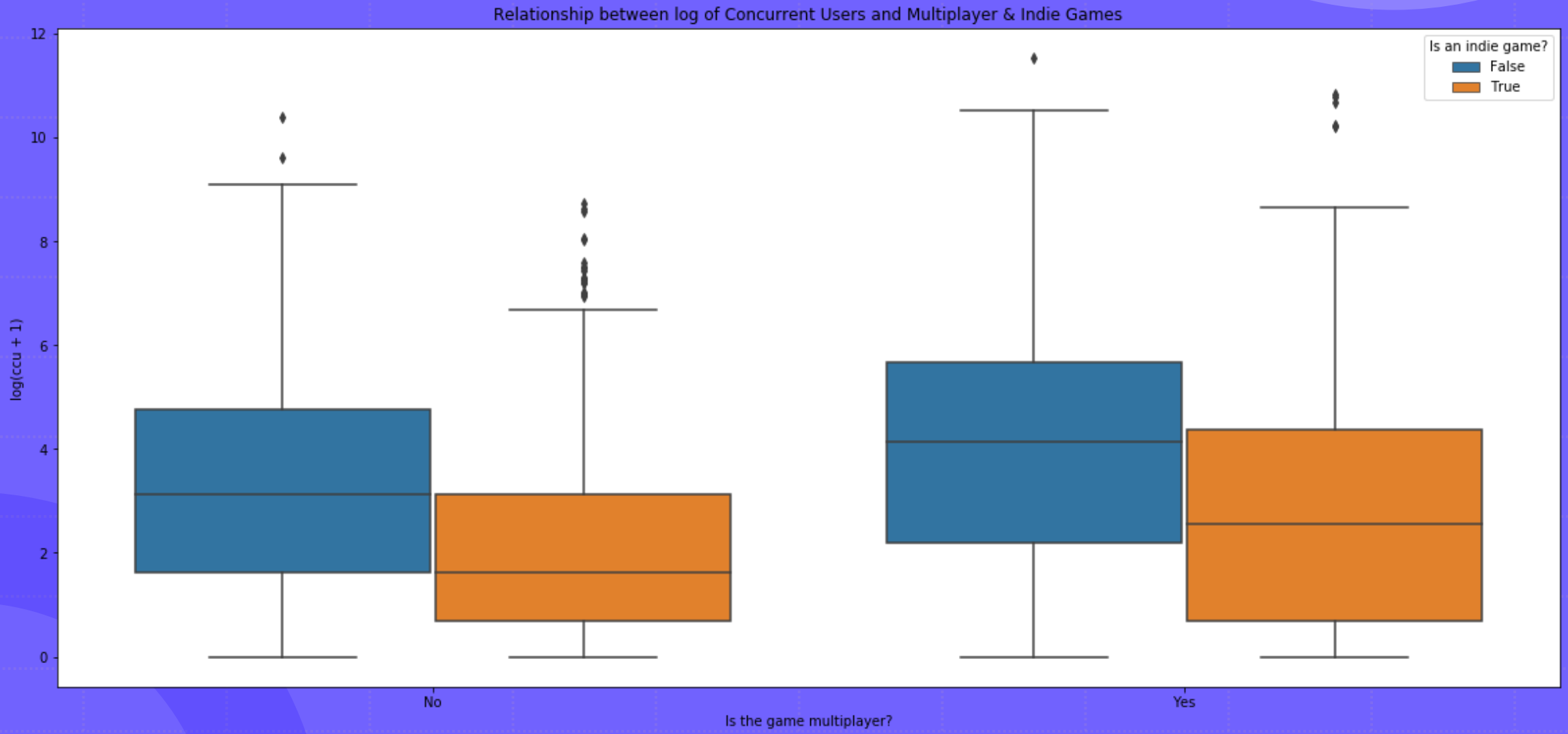
```
[25]  df["ccu_log"] = np.log(df["ccu"])
      ✓ 0.5s                                                    Python
```

```
C:\Users\Vuc\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.8_qbz5n2kfra8p0\LocalCache\local-packages\Python38\site-packages\pandas\core\arraylike.py:358: RuntimeWarning: divide by zero encountered in log
  result = getattr(ufunc, method)(*inputs, **kwargs)
```

```
[27]  df["ccu_log"] = np.log(df["ccu"] + 1)
      df["ccu_log"]
      ✓ 0.3s                                                    Python
```

```
0       9.617071
1       4.488636
2       4.672829
3       1.791759
4       4.779123
          ...
```

Descriptive Analytics



Descriptive Analytics

Showing the mean log CCus by multiplayer and indie games

		ccu_log
is_multiplayer	**is_indie**	
False	False	3.294803
	True	2.136801
True	False	4.117114
	True	2.889345

Showing the median log CCus by multiplayer and indie games

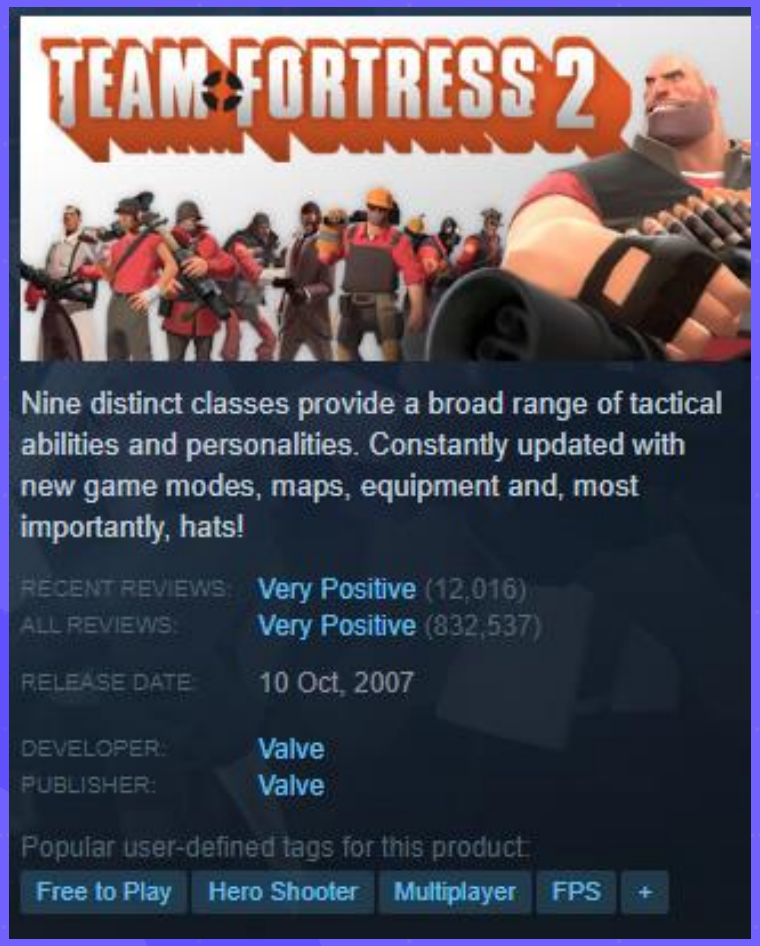
		ccu_log
is_multiplayer	**is_indie**	
False	False	3.135494
	True	1.609438
True	False	4.143135
	True	2.564949

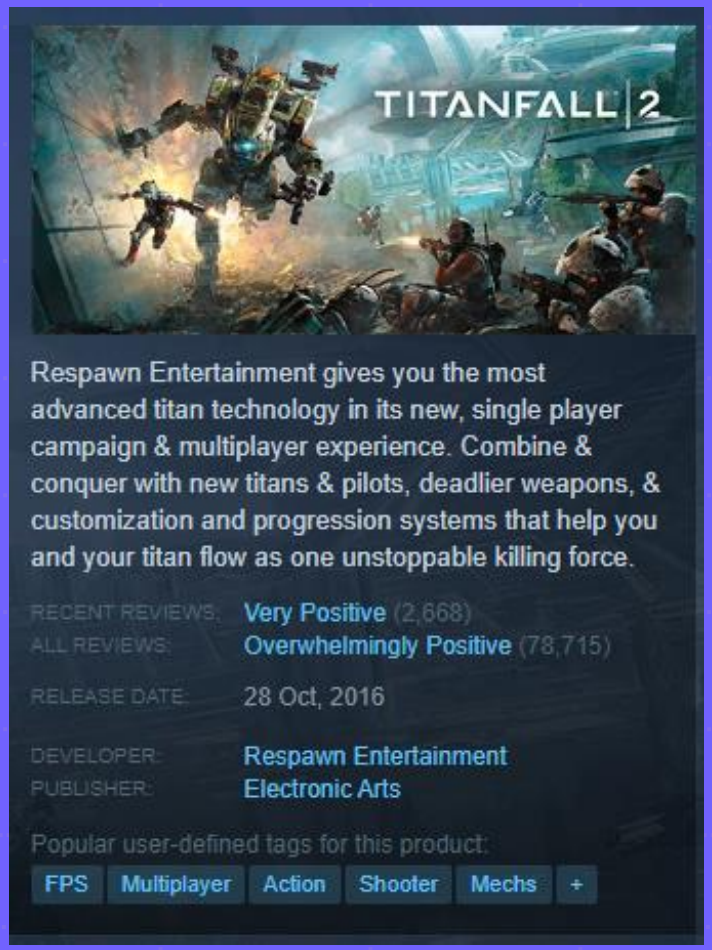
Showing the std of log CCus by multiplayer and indie games

		ccu_log
is_multiplayer	**is_indie**	
False	False	2.202848
	True	1.844793
True	False	2.427851
	True	2.493471

Inference

Free-to-play vs. Paid Games





Inference

Relive the Capcom classics! Get 1943 -The Battle of Midway- and game logo wallpapers free with your download!

RECENT REVIEWS: Mostly Positive (434)
ALL REVIEWS: Mixed (1,788)

RELEASE DATE: May 24, 2021

DEVELOPER: CAPCOM Co., Ltd.
PUBLISHER: CAPCOM Co., Ltd.

Popular user-defined tags for this product:

Action | Arcade | 2D Platformer | 2D Fighter | 2D | +

EMBARK ON A JOURNEY TO OWN IT ALL! Build your empire on a colorful and 3D lively city animated by funny sidekicks! MONOPOLY® PLUS brings the classic franchise to a new level on PC with amazing animations, customizable house rules!

RECENT REVIEWS: Mixed (292)
ALL REVIEWS: Mixed (7,517)

RELEASE DATE: Sep 7, 2017

DEVELOPER: Ubisoft Pune
PUBLISHER: Ubisoft

Popular user-defined tags for this product:

Multiplayer | Board Game | Casual | Tabletop | +

Explore a thrilling, open-world MMO filled with danger and opportunity where you'll forge a new destiny on the supernatural island of Aeternum.

RECENT REVIEWS: Mixed (32,887)
ALL REVIEWS: Mixed (177,941)

DATE: 28 Sep, 2021

ER: Amazon Games
R: Amazon Games

ser-defined tags for this product:

y Multiplayer | Open World | MMORPG | +

The highest rated* annual sports title returns with NBA 2K18, featuring unparalleled authenticity and improvements on the court.*According to 2008 - 2016 Metacritic.com

RECENT REVIEWS: Mixed (15)
ALL REVIEWS: Mostly Negative (14,914)

RELEASE DATE: 14 Sep, 2017

DEVELOPER: Visual Concepts
PUBLISHER: 2K

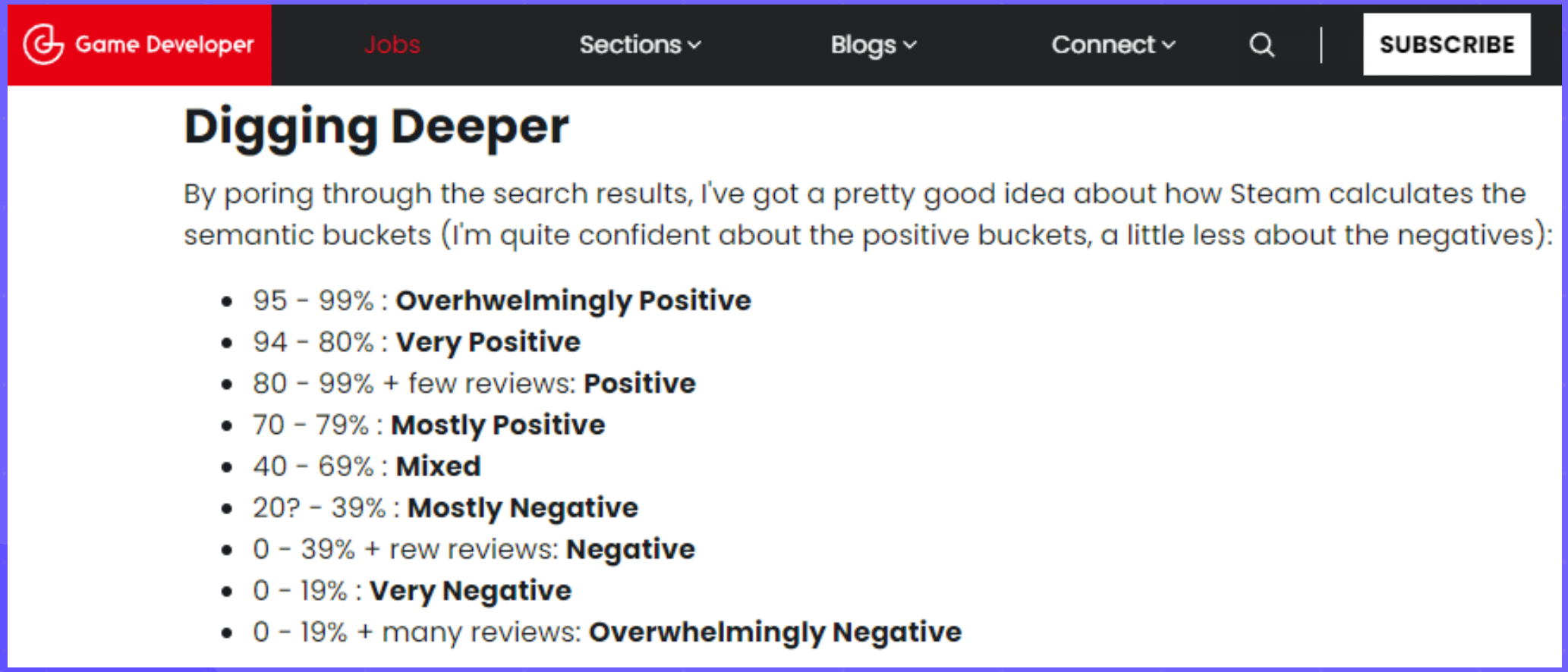
Popular user-defined tags for this product:

Sports | Basketball | Gambling | Multiplayer | +

Inference


STAR RATINGS
|< < PREV RANDOM NEXT > >|
UNDERSTANDING ONLINE STAR RATINGS:
[HAS ONLY ONE REVIEW]
EXCELLENT
OK
CRAP
|< < PREV RANDOM NEXT > >|

Inference


Game Developer
Jobs
Sections
Blogs
Connect
SUBSCRIBE
Digging Deeper
By poring through the search results, I've got a pretty good idea about how Steam calculates the semantic buckets (I'm quite confident about the positive buckets, a little less about the negatives):
95 - 99% : Overhwelmingly Positive
94 - 80% : Very Positive
80 - 99% + few reviews: Positive
70 - 79% : Mostly Positive
40 - 69% : Mixed
20? - 39% : Mostly Negative
0 - 39% + rew reviews: Negative
0 - 19% : Very Negative
0 - 19% + many reviews: Overwhelmingly Negative

Inference

Do free-to-play games get rated more positively than paid games?

$$H_0 : p_{\text{positively rated free games}} - p_{\text{positively rated paid games}} = 0$$

$$H_A : p_{\text{positively rated free games}} - p_{\text{positively rated paid games}} \neq 0$$

3.3 Setup

First, we need to decide on a threshold on whether a game is negatively-reviewed or not. 0.5 is certainly not it in this case- if half the pl
labels these games as Mixed; I will consider Mixed as a negative review proportion).

Looking it up only gave me this old result, but I believe we can use it. It seems that 'Mostly Positive' starts at 70%, so we will let that be t

```
df["percentage_positive_reviews"] = df["positive_reviews"] / (df["positive_reviews"] + df["negative_reviews"])

df["is_positively_reviewed"] = df["percentage_positive_reviews"] >= 0.69
df["is_positively_reviewed"] = df["is_positively_reviewed"].map({True: 1, False: 0})

sample_free = df[df["price"] == 0]
sample_paid = df[df["price"] != 0]
```

[9] ✓ 0.4s

```
prop_free_positive = np.mean(sample_free["is_positively_reviewed"])
prop_paid_positive = np.mean(sample_paid["is_positively_reviewed"])

prop_free_positive, prop_paid_positive
```

[10] ✓ 0.3s

... (0.6620689655172414, 0.7900826446280992)

3.4 Test Statistic

We can calculate the test statistic for this hypothesis test.

```
sample_diff = prop_free_positive - prop_paid_positive
sample_std = ((prop_free_positive * (1 - prop_free_positive) / sample_free.shape[0])
              + (prop_paid_positive * (1 - prop_paid_positive) / sample_paid.shape[0]))**0.5

z_stat = (sample_diff - 0) / sample_std
z_stat
```

1] ✓ 0.3s

. -4.2469663193292195

```
from scipy.stats import norm

p_value = 2 * norm.cdf(z_stat)
p_value
```

2] ✓ 0.3s

. 2.1668451810872935e-05

Inference

```
from scipy.stats import norm

p_value = 2 * norm.cdf(z_stat)
p_value
```

✓ 0.3s

```
2.1668451810872935e-05
```

$$H_A : p_{\text{positively rated free games}} - p_{\text{positively rated paid games}} \neq 0$$

```
prop_free_positive, prop_paid_positive
```

✓ 0.3s

```
(0.6620689655172414, 0.7900826446280992)
```

Linear Regression

Is there a relationship between a game's concurrent users (CCU) and:

a. Owner count
b. Release date (measured by year)
c. Positive & negative reviews
d. Median playtime (in 2 weeks & all-time)
e. Multiplayer games & indie games

Linear Regression

```
df[["positive_reviews", "negative_reviews"]].corr()
```

✓ 0.3s

	positive_reviews	negative_reviews
positive_reviews	1.00000	0.74537
negative_reviews	0.74537	1.00000

```
df["percentage_positive_reviews"] = df["positive_reviews"] / (df["positive_reviews"] + df["negative_reviews"])

df.head()[["name", "positive_reviews", "negative_reviews", "percentage_positive_reviews"]]
```

✓ 0.4s Python

	name	positive_reviews	negative_reviews	percentage_positive_reviews
0	The Adventure Pals	1205	98	0.924789
1	Command & Conquer 4: Tiberian Twilight	531	2465	0.177236
2	SNOW - The Ultimate Edition	7942	4153	0.656635
3	Back to Bed	1957	451	0.812708
4	Dude Simulator 2	887	785	0.530502

Linear Regression

Use all variables without transformation

OLS Regression Results

Dep. Variable:	ccu	R-squared:	0.993
Model:	OLS	Adj. R-squared:	0.992
Method:	Least Squares	F-statistic:	1491.
Date:	Wed, 08 Dec 2021	Prob (F-statistic):	5.57e-138
Time:	15:21:42	Log-Likelihood:	-1179.6
No. Observations:	148	AIC:	2387.
Df Residuals:	134	BIC:	2429.
Df Model:	13		
Covariance Type:	nonrobust		

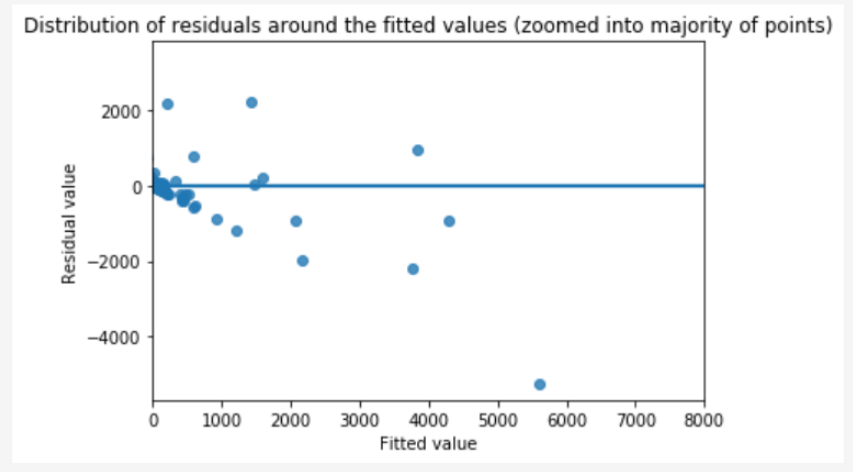
Linear Regression

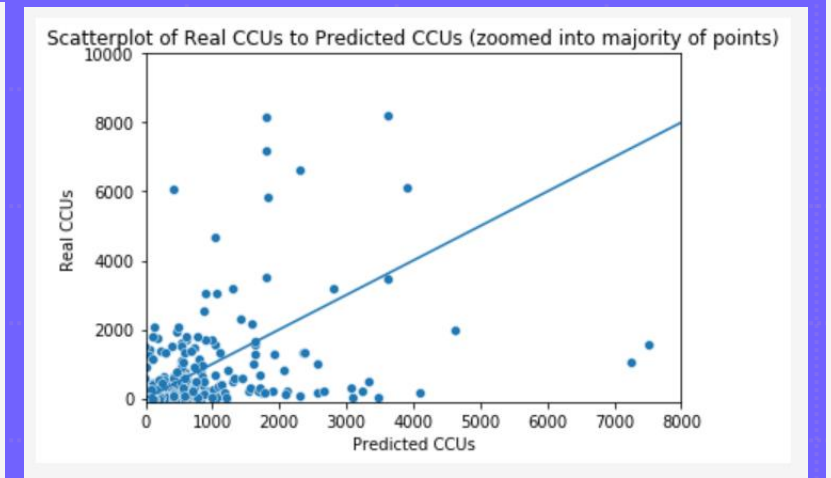
Use all variables without transformation

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	6843.5549	3.92e+04	0.175	0.862	-7.07e+04	8.44e+04
owners[T.200,000 - 500,000]	86.0885	147.551	0.583	0.561	-205.742	377.919
owners[T.500,000 - 1,000,000]	-68.1209	188.901	-0.361	0.719	-441.734	305.492
owners[T.1,000,000 - 2,000,000]	477.3368	248.013	1.925	0.056	-13.190	967.864
owners[T.2,000,000 - 5,000,000]	-1293.2069	365.858	-3.535	0.001	-2016.809	-569.605
owners[T.5,000,000 - 10,000,000]	1115.6334	573.571	1.945	0.054	-18.789	2250.056
owners[T.10,000,000 - 20,000,000]	3154.1664	538.740	5.855	0.000	2088.633	4219.700
owners[T.20,000,000 - 50,000,000]	1.006e+05	804.035	125.098	0.000	9.9e+04	1.02e+05
owners[T.50,000,000 - 100,000,000]	-1.5e-11	1.49e-10	-0.101	0.920	-3.1e-10	2.8e-10
owners[T.100,000,000 - 200,000,000]	2.211e-11	2.05e-11	1.077	0.283	-1.85e-11	6.27e-11
is_multiplayer[T.True]	133.3694	138.160	0.965	0.336	-139.888	406.627
is_indie[T.True]	38.7132	131.592	0.294	0.769	-221.552	298.978
release_year	-3.3969	19.468	-0.174	0.862	-41.901	35.107
percentage_positive_reviews	-113.6836	388.521	-0.293	0.770	-882.110	654.743
median_forever	0.0709	0.054	1.314	0.191	-0.036	0.178
median_2weeks	2.6022	0.157	16.606	0.000	2.292	2.912

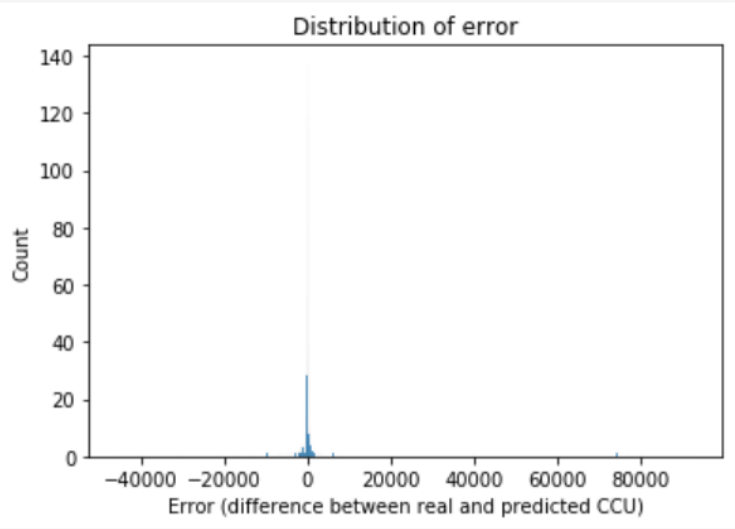
Linear Regression

Visualize the residuals & fitted values



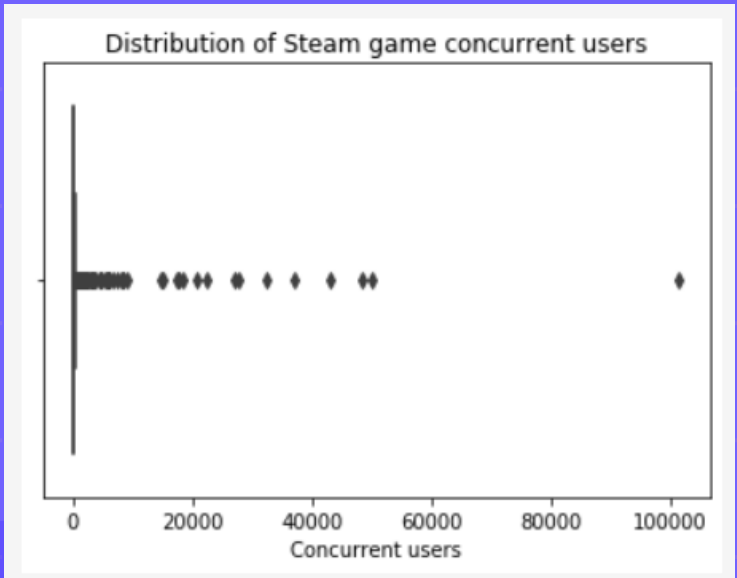


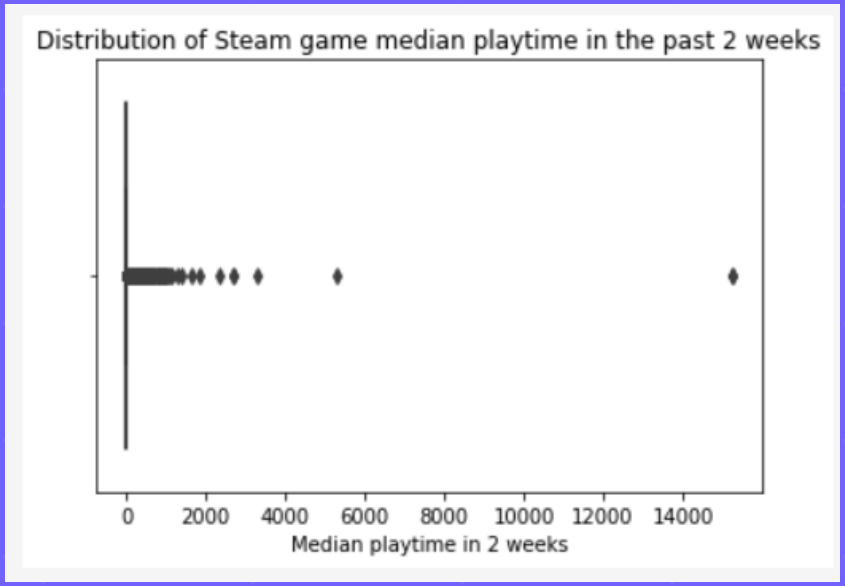

```
Mean error: 109.4112209666126
```

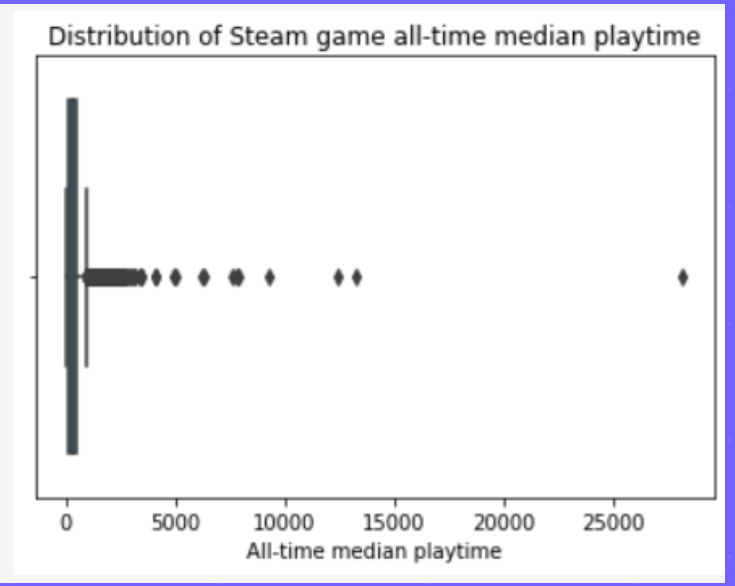


Linear Regression

Visualize the responsible numerical variables

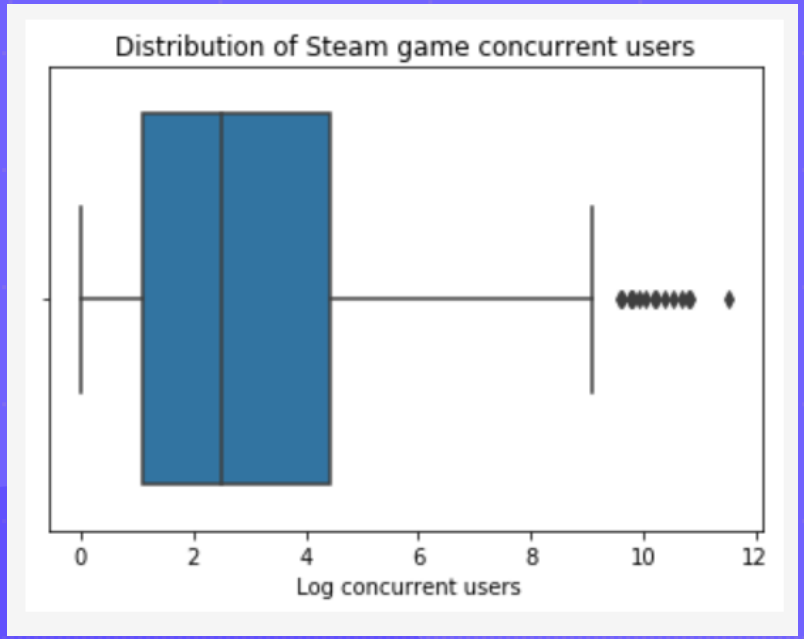


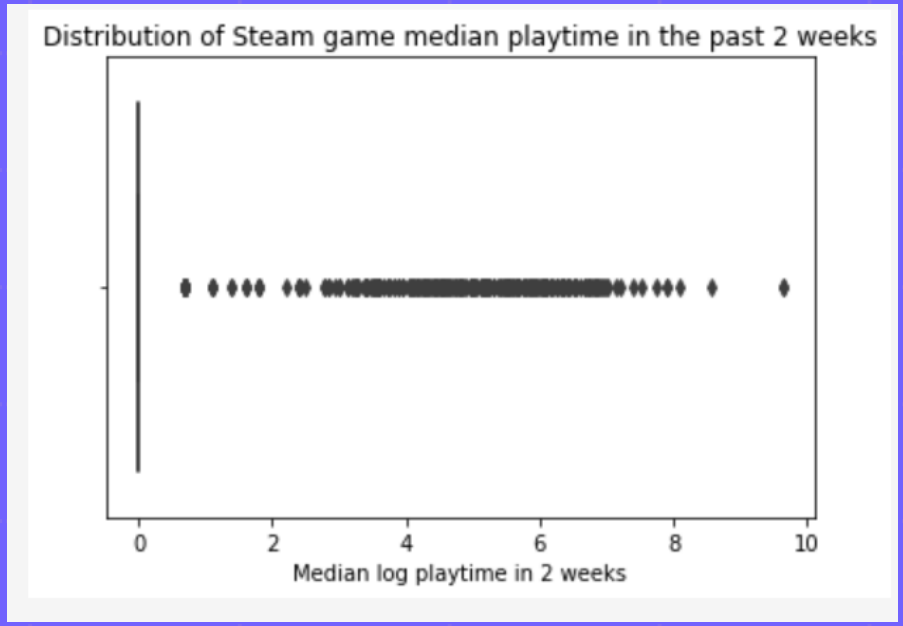


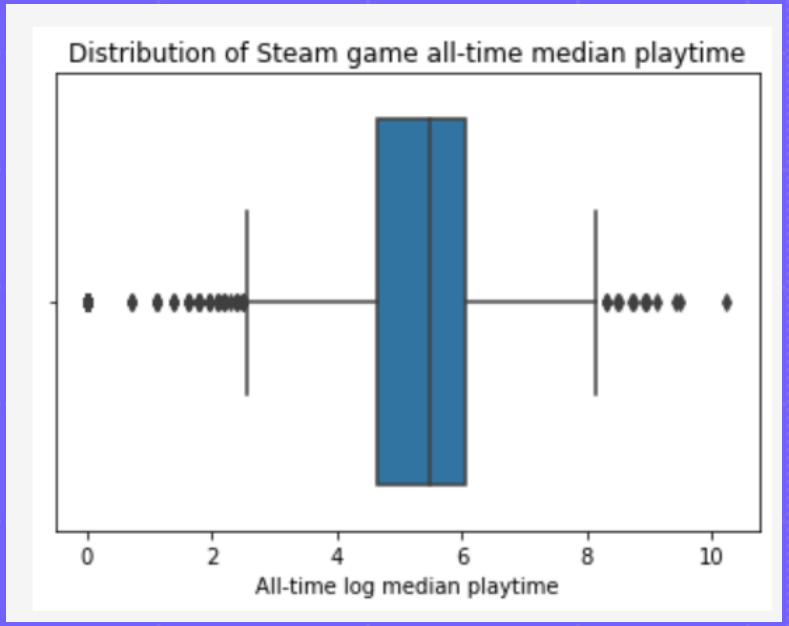


Linear Regression

Post-transformation visualization







Linear Regression

```
ols_log_model.summary()
```

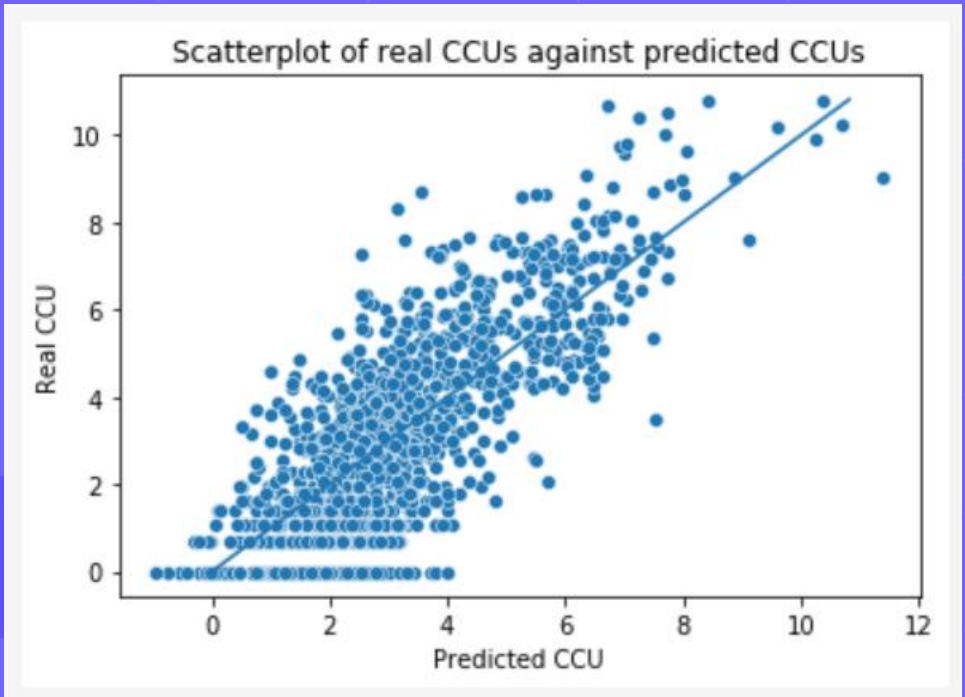
OLS Regression Results

Dep. Variable:	ccu_log	R-squared:	0.663
Model:	OLS	Adj. R-squared:	0.630
Method:	Least Squares	F-statistic:	20.27
Date:	Wed, 08 Dec 2021	Prob (F-statistic):	1.24e-25
Time:	15:21:50	Log-Likelihood:	-252.10
No. Observations:	148	AIC:	532.2
Df Residuals:	134	BIC:	574.2
Df Model:	13		
Covariance Type:	nonrobust		

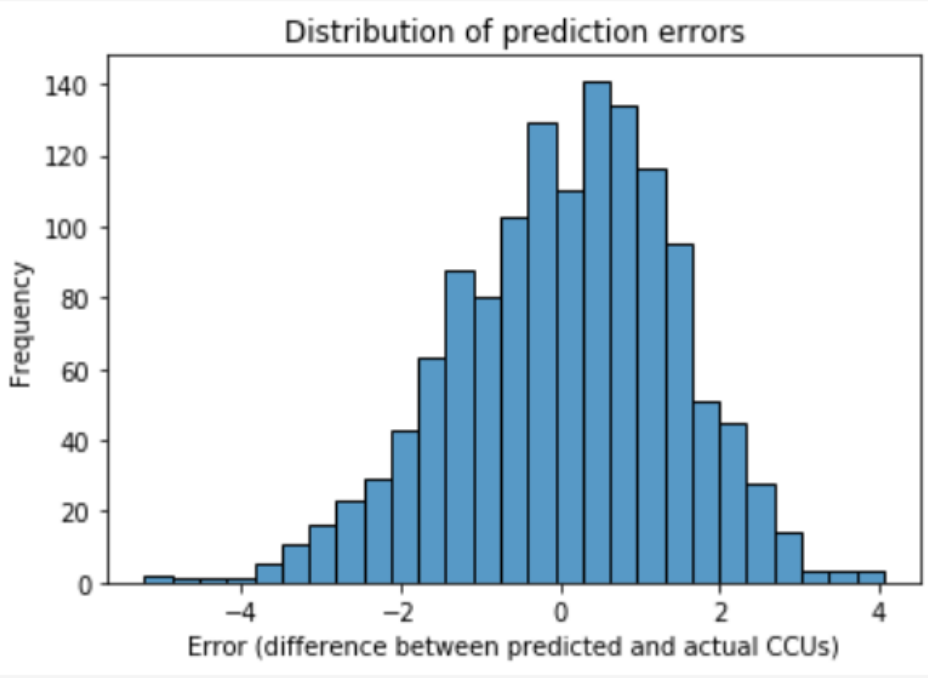
	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-249.3690	77.146	-3.232	0.002	-401.950	-96.788
owners[T.200,000 - 500,000]	0.6191	0.283	2.186	0.031	0.059	1.179
owners[T.500,000 - 1,000,000]	1.2681	0.365	3.477	0.001	0.547	1.990
owners[T.1,000,000 - 2,000,000]	1.8378	0.496	3.705	0.000	0.857	2.819
owners[T.2,000,000 - 5,000,000]	0.9767	0.695	1.404	0.162	-0.399	2.352
owners[T.5,000,000 - 10,000,000]	1.9358	1.007	1.923	0.057	-0.055	3.927
owners[T.10,000,000 - 20,000,000]	3.3979	1.133	2.999	0.003	1.157	5.639
owners[T.20,000,000 - 50,000,000]	5.8514	1.511	3.874	0.000	2.864	8.839
owners[T.50,000,000 - 100,000,000]	-1.055e-15	2.66e-15	-0.396	0.693	-6.33e-15	4.22e-15
owners[T.100,000,000 - 200,000,000]	-2.674e-16	4.24e-15	-0.063	0.950	-8.66e-15	8.12e-15
is_multiplayer[T.True]	0.4099	0.262	1.567	0.119	-0.107	0.927
is_indie[T.True]	-0.8222	0.251	-3.282	0.001	-1.318	-0.327
release_year	0.1231	0.038	3.213	0.002	0.047	0.199
percentage_positive_reviews	3.0579	0.741	4.126	0.000	1.592	4.524
median_forever_log	0.2307	0.077	2.996	0.003	0.078	0.383
median_2weeks_log	0.3706	0.091	4.089	0.000	0.191	0.550

Linear Regression

Performance (General)




```
Mean error: 0.05495174616303576
```



Linear Regression

Performance (Specific Games)

	name	developer	owners	ccu
148	Terraria	Re-Logic	20,000,000 - 50,000,000	48257
738	Sid Meier's Civilization V	Firaxis Games, Aspyr (Mac), Aspyr (Linux)	5,000,000 - 10,000,000	18425
472	Borderlands 2	Gearbox Software, Aspyr (Mac), Aspyr (Linux)	10,000,000 - 20,000,000	3361
1299	OneShot	Future Cat LLC	500,000 - 1,000,000	126
1450	Just Shapes & Beats	Berzerk Studio	500,000 - 1,000,000	121

Linear Regression

Performance (Specific Games)

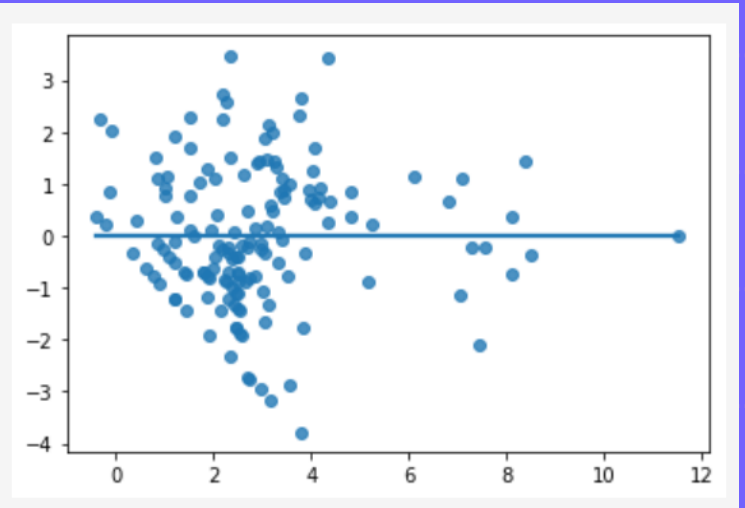
'ccu' stands for concurrent users, 'log' means the data went through a logarithm transformation

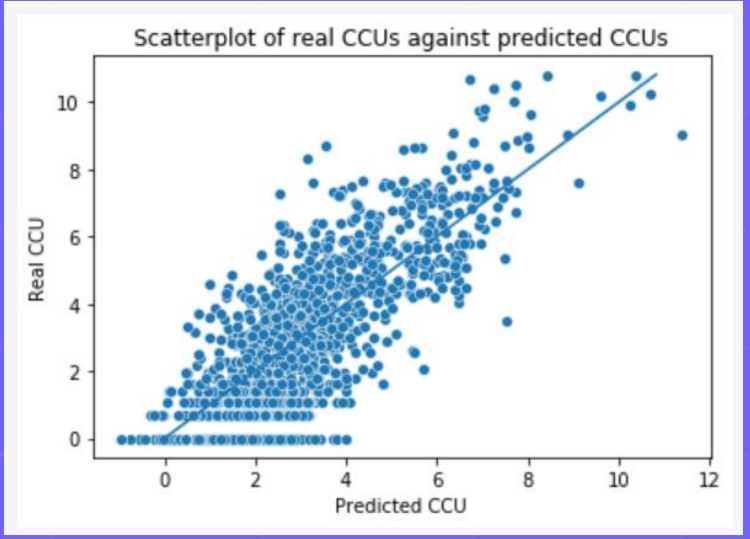
	real_ccu_log	predicted_ccu_log	real_ccu	predicted_ccu	error	percentage_error
Terraria	10.784317	10.380700	48257	32230.526842	16026.473158	33.2%
Sid Meier's Civilization V	9.821518	8.384677	18425	4378.442470	14046.557530	76.2%
Borderlands 2	8.120291	8.486470	3361	4847.717605	1486.717605	44.2%
OneShot	4.844187	5.859866	126	349.677262	223.677262	177.5%
Just Shapes & Beats	4.804021	6.241859	121	512.812721	391.812721	323.8%

Linear Regression

Conclusion

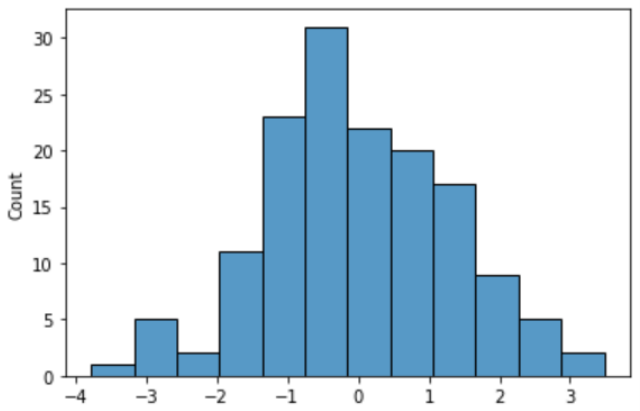
$$\overline{\log(\text{Concurrent Users} + 1)} = -249.4 + 0.41\,[\text{is\_multiplayer}] - 0.82\,[\text{is\_indie}] + 3.1\,[\text{percentage\_positive\_reviews}] + 0.12\,[\text{release\_year}] + 0.23\log([\text{median\_forever}] + 1) + 0.37\log([\text{median\_2weeks}] + 1) + 0.62\,[\text{200,000 - 500,000 owners}] + 1.3\,[\text{500,000 - 1,000,000 owners}] + 1.8\,[\text{1,000,000 - 2,000,000 owners}] + 0.98\,[\text{2,000,000 - 5,000,000 owners}] + 1.94\,[\text{5,000,000 - 10,000,000 owners}] + 3.4\,[\text{10,000,000 - 20,000,000 owners}] + 5.85\,[\text{20,000,000 - 50,000,000 owners}] - 2.7 \times 10^{-15}\,[\text{50,000,000 - 100,000,000 owners}] - 2.7 \times 10^{-16}\,[\text{100,000,000 - 200,000,000 owners}]$$

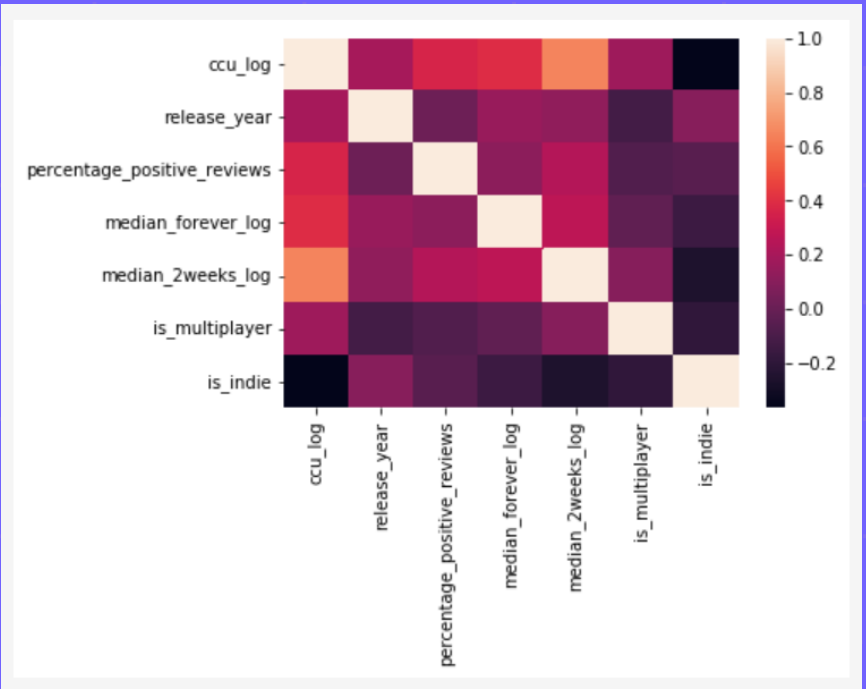





```
Mean residuals: -6.420997367440362e-13

<matplotlib.axes._subplots.AxesSubplot at 0x22c0d19d448>
```





Logistic Regression

Is there a relationship between the log-odds that a game is positive-reviewed and:

a. Price

b. Median playtime (in 2 weeks & all-time)

c. Multiplayer games & indie games

Logistic Regression

```
logit_model.summary()
```

Logit Regression Results

Dep. Variable:	is_negatively_reviewed	No. Observations:	150
Model:	Logit	Df Residuals:	143
Method:	MLE	Df Model:	6
Date:	Wed, 08 Dec 2021	Pseudo R-squ.:	0.1393
Time:	16:25:56	Log-Likelihood:	-71.144
converged:	True	LL-Null:	-82.662
Covariance Type:	nonrobust	LLR p-value:	0.0007846

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.6412	0.812	-0.790	0.429	-2.232	0.949
is_multiplayer[T.True]	0.3331	0.461	0.722	0.470	-0.571	1.237
is_indie[T.True]	0.5227	0.459	1.139	0.255	-0.377	1.422
median_forever_log	0.0785	0.131	0.598	0.550	-0.179	0.336
median_2weeks_log	0.1177	0.310	0.379	0.705	-0.491	0.726
price	-0.0446	0.029	-1.518	0.129	-0.102	0.013
ccu_log	-0.4268	0.150	-2.852	0.004	-0.720	-0.133

Logistic Regression

Parsimoniousness – Backwards Elimination

```
def backwards_elim_bic(vars, initial_bic, iter = 1):
    lowest_bic = initial_bic
    removed_feature = "None"
    print("Iteration " + str(iter))

    for i in range(len(vars)):

        # Create a set of variables that exclude the i-th explanatory variable and create a formula from that
        test_vars = [vars[j] for j in range(len(vars)) if j != i]
        test_formula = "is_positively_reviewed ~ " + (" + ".join(test_vars))

        # Fit the model to that formula and get the adjusted r-squared value
        test_model = smf.ols(data=train, formula=test_formula).fit()
        test_bic = test_model.bic

        print("By removing " + vars[i] + ", BIC = " + str(test_bic))

        # If the adjusted r-squared value is greater than the current largest, then replace it
        if test_bic < lowest_bic:
            removed_feature = vars[i]
            lowest_bic = test_bic

    # Print the resulting variable to remove
    print("Initial BIC  = " + str(initial_bic) + "\nLowest BIC = " +
          str(lowest_bic) + " achieved by removing " + removed_feature)

      # Recursively run until adjusted R squared does not increase
    if removed_feature != "None":

        resultant_vars = [i for i in vars if i != removed_feature]

        if len(resultant_vars) == 1:
            print("Left with one variable (" + resultant_vars[0] + "), algorithm halted")
        else:
            backwards_elim_bic(resultant_vars, lowest_bic, iter + 1)
```

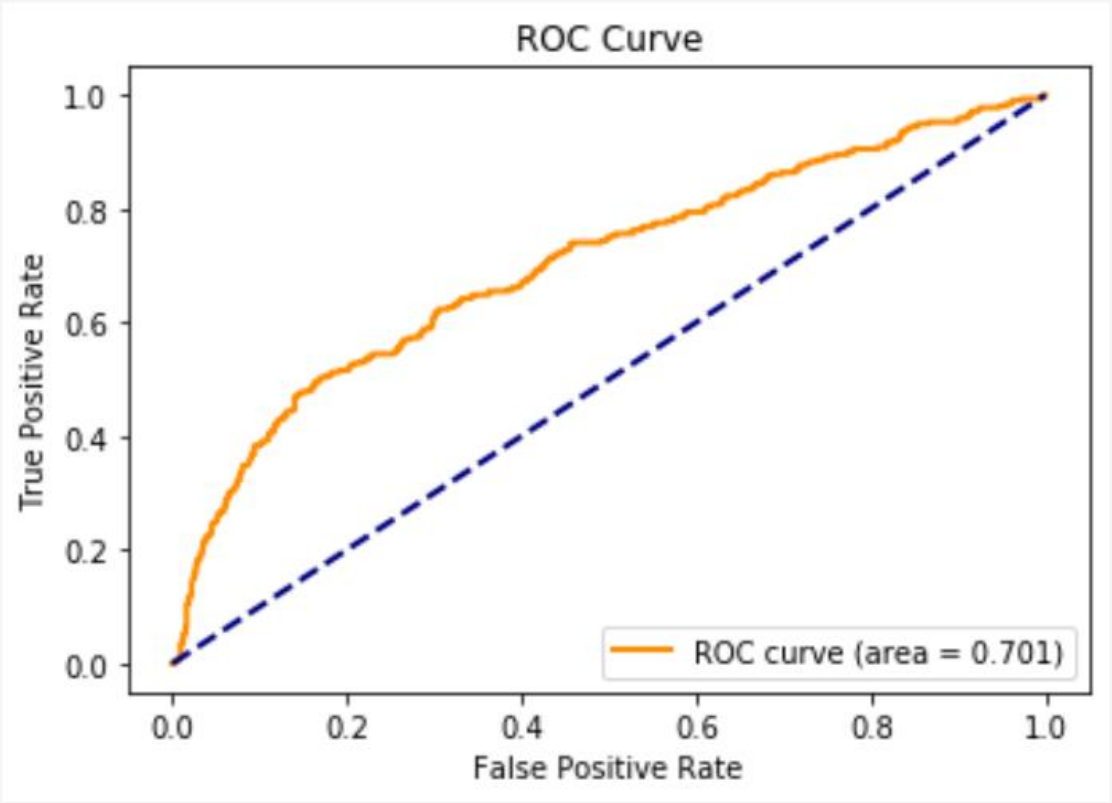
Logistic Regression

Parsimoniousness – Backwards Elimination

```
Iteration 1
By removing median_forever_log, BIC = 180.66785411637497
By removing median_2weeks_log, BIC = 181.85901030775332
By removing price, BIC = 182.3274832757041
By removing ccu_log, BIC = 190.12170596335952
By removing is_multiplayer, BIC = 181.29178237910182
By removing is_indie, BIC = 181.07108402429483
Initial BIC  = 177.36278836338792
Lowest BIC = 177.36278836338792 achieved by removing None
```

Logistic Regression

ROC Curve & AUC, finding the threshold



	threshold	tpr	fpr
0	0.35	0.525316	0.205996

	threshold	tpr	fpr
0	0.36	0.506329	0.180851

	threshold	tpr	fpr
0	0.37	0.496835	0.167311

	threshold	tpr	fpr
0	0.38	0.481013	0.15764

	threshold	tpr	fpr
0	0.39	0.455696	0.140232

Logistic Regression

Apply threshold to test set

```
tpr_test, fpr_test
```

✓ 0.4s

```
(0.1155555555555555, 0.6274074074074074)
```

That is quite bad.

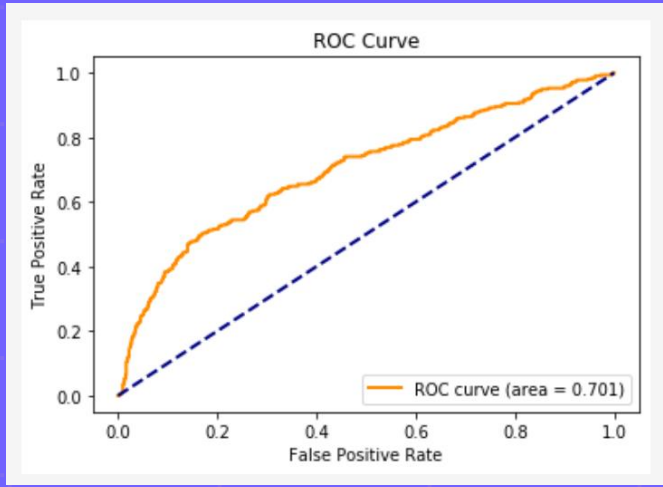
Logistic Regression

Conclusion

Logit Regression Results

Dep. Variable:	is_negatively_reviewed	No. Observations:	150
Model:	Logit	Df Residuals:	143
Method:	MLE	Df Model:	6
Date:	Wed, 08 Dec 2021	Pseudo R-squ.:	0.1393
Time:	16:25:56	Log-Likelihood:	-71.144
converged:	True	LL-Null:	-82.662
Covariance Type:	nonrobust	LLR p-value:	0.0007846

	coef	std err	z	P>|z|	[0.025	0.975]
Intercept	-0.6412	0.812	-0.790	0.429	-2.232	0.949
is_multiplayer[T.True]	0.3331	0.461	0.722	0.470	-0.571	1.237
is_indie[T.True]	0.5227	0.459	1.139	0.255	-0.377	1.422
median_forever_log	0.0785	0.131	0.598	0.550	-0.179	0.336
median_2weeks_log	0.1177	0.310	0.379	0.705	-0.491	0.726
price	-0.0446	0.029	-1.518	0.129	-0.102	0.013
ccu_log	-0.4268	0.150	-2.852	0.004	-0.720	-0.133



	threshold	tpr	fpr
0	0.35	0.525316	0.205996

	threshold	tpr	fpr
0	0.36	0.506329	0.180851

	threshold	tpr	fpr
0	0.37	0.496835	0.167311

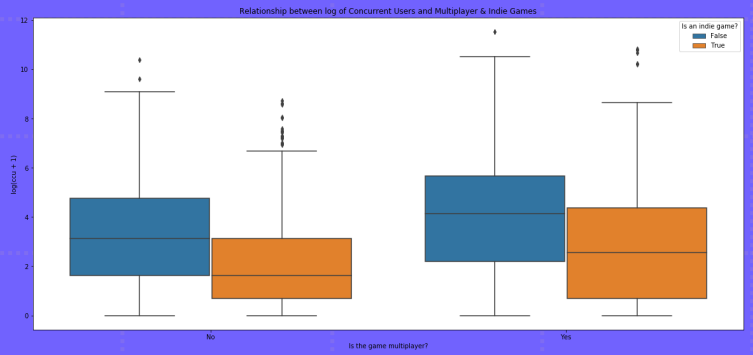
	threshold	tpr	fpr
0	0.38	0.481013	0.15764

	threshold	tpr	fpr
0	0.39	0.455696	0.140232

That is quite bad.

Conclusion

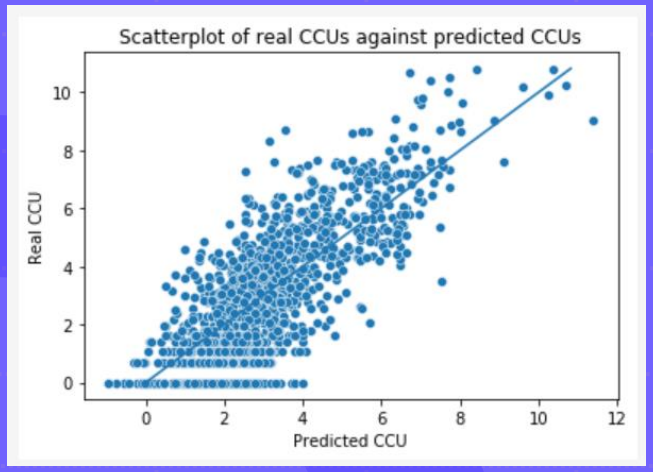
Descriptive Analytics



Inference

Rejected null; free games are more positively-reviewed than paid games

Linear Regression



owners[T.100,000,000 - 200,000,000]
is_multiplayer[T.True]
is_indie[T.True]
release_year
percentage_positive_reviews
median_forever_log
median_2weeks_log

Logistic Regression