Do Streams Sell?

A Linear Regression by Nick Sherwin

Objective

To understand the relationship between unit sales of a video game title and streaming metrics from Twitch.

Methodology

Data Sources & References:

- VGChartz
- SullyGnome

Primary Tools:

- Pandas, Beautiful Soup, StatsModels, Scikit-learn

Methodology

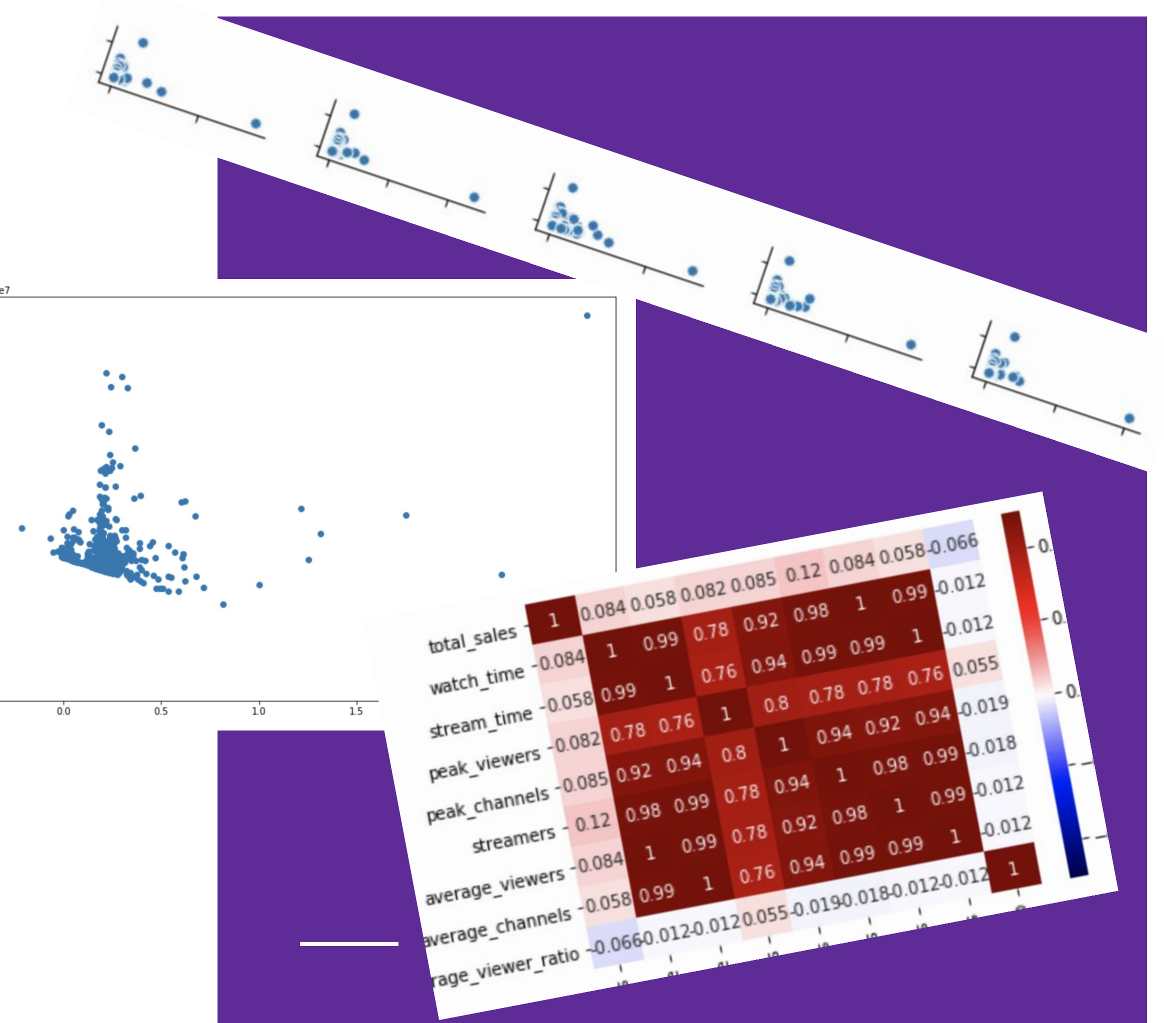
Scope

- 2016 - Present
- Global Sales + Streams
- All Platforms

Caveats:

- Excluding Free-to-Play (F2P)
- Excluding Micro-Transactions
- Streams w/ Only 10+ Viewers

Preliminary Results



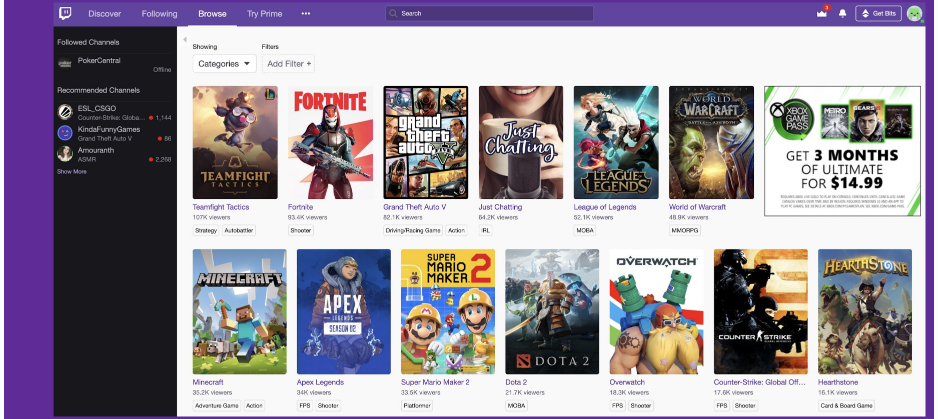
total_sales	1	0.084	0.058	0.082	0.085	0.12	0.084	0.058	-0.066
watch_time	0.084	1	0.99	0.78	0.92	0.98	1	0.99	-0.012
stream_time	0.058	0.99	1	0.76	0.94	0.99	0.99	1	-0.012
peak_viewers	0.082	0.78	0.76	1	0.8	0.78	0.78	0.76	0.055
peak_channels	0.085	0.92	0.94	0.8	1	0.94	0.92	0.94	-0.019
streamers	0.12	0.98	0.99	0.78	0.94	1	0.98	0.99	-0.018
average_viewers	0.084	1	0.99	0.78	0.92	0.98	1	0.99	-0.012
average_channels	0.058	0.99	1	0.76	0.94	0.99	0.99	1	-0.012
rage_viewer_ratio	-0.066	-0.012	-0.012	0.055	-0.019	-0.018	-0.012	-0.012	1

But Wait...

Peak_Channels

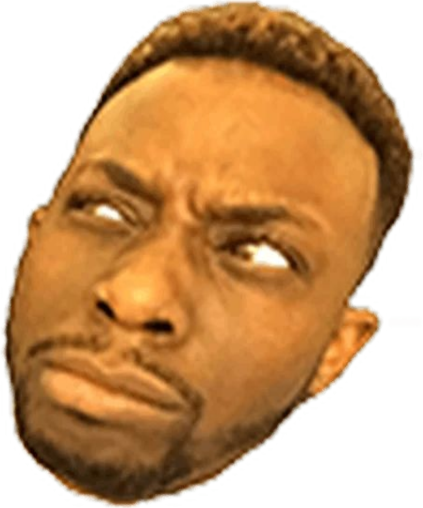
447.76	2016 - 2018 (All Games)
590.53	2016 - 2018 (Top Games)
1053.68	2018 (Top Games)
1542.79	2018 (Monthly)

Recommendation



- Platform Sponsorship
- Game Dev Dashboard
- Creator First

Future Study



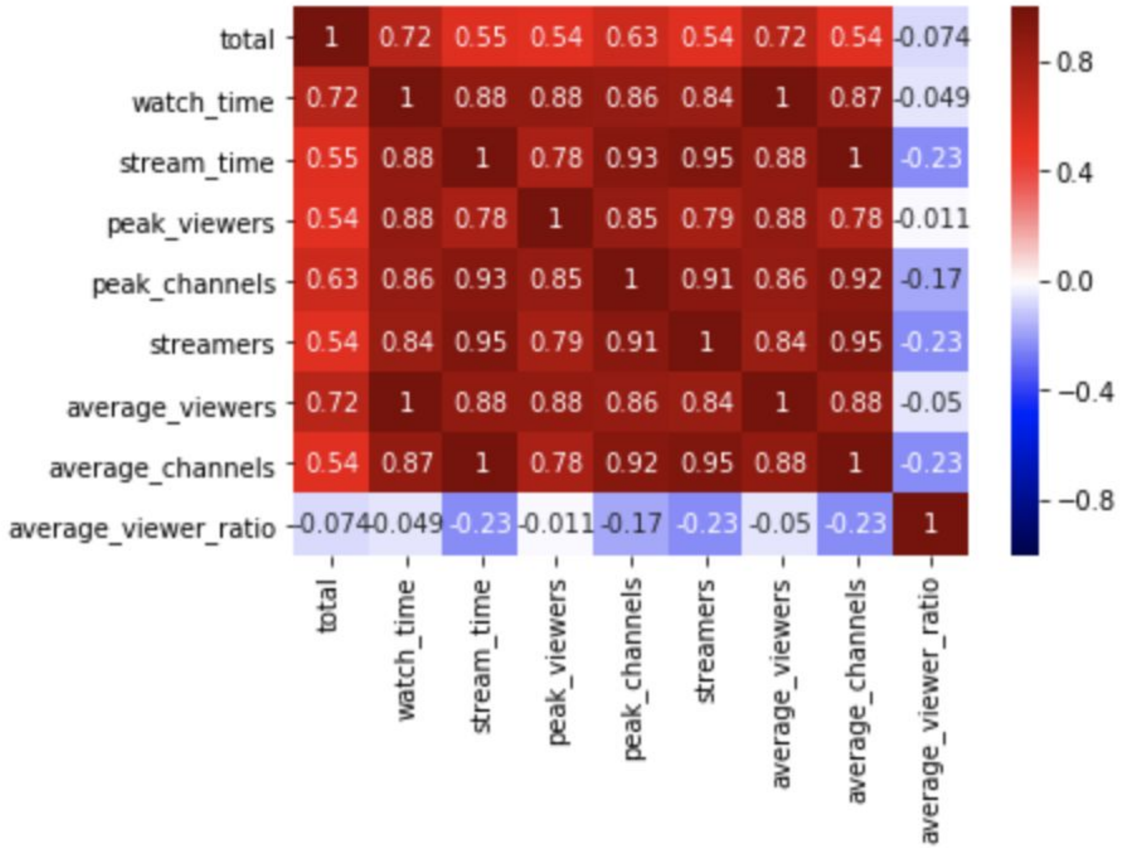
- Twitch API
- Build Dev Dashboard
- Chat in Stream

Questions?

Appendix:
Monthly Model

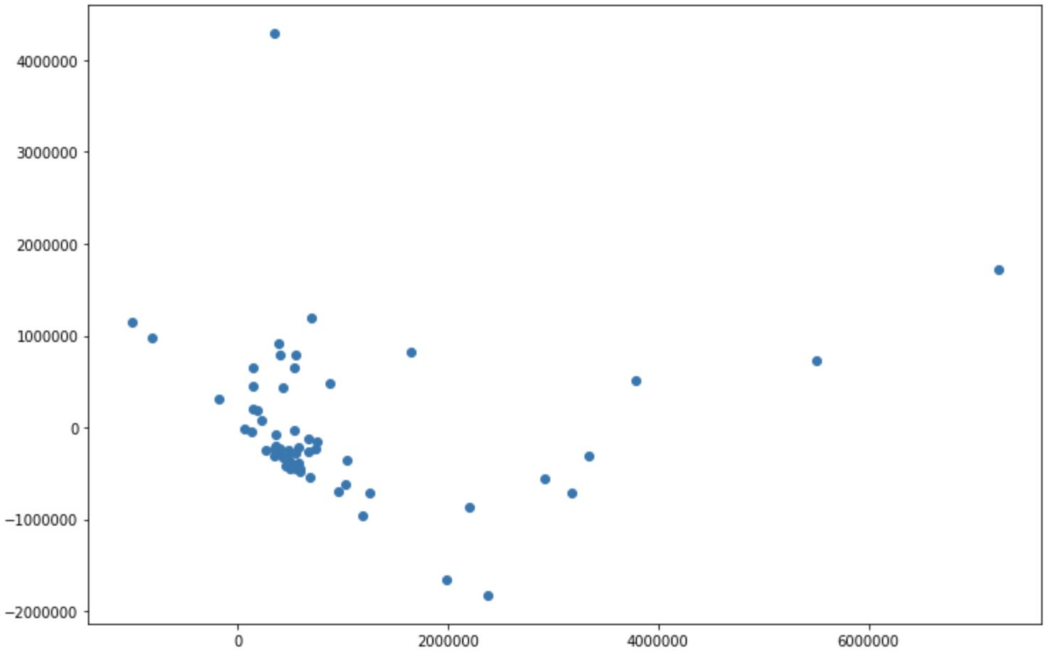
Adjusted R2	.693
Peak Channels Coef	1542
Cross Validated Score	0.551
Test Score	.756

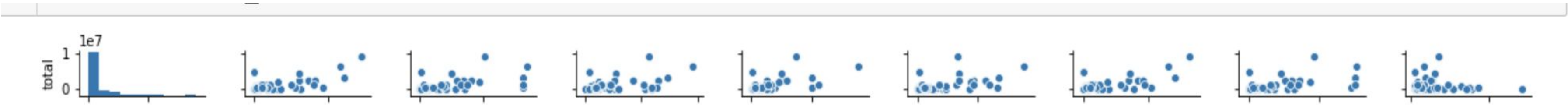
Appendix: Monthly Correlation



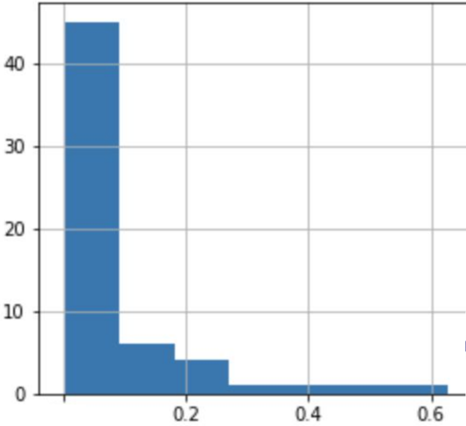
	total	watch_time	stream_time	peak_viewers	peak_channels	streamers	average_viewers	average_channels	average_viewer_ratio
total	1	0.72	0.55	0.54	0.63	0.54	0.72	0.54	-0.074
watch_time	0.72	1	0.88	0.88	0.86	0.84	1	0.87	-0.049
stream_time	0.55	0.88	1	0.78	0.93	0.95	0.88	1	-0.23
peak_viewers	0.54	0.88	0.78	1	0.85	0.79	0.88	0.78	-0.011
peak_channels	0.63	0.86	0.93	0.85	1	0.91	0.86	0.92	-0.17
streamers	0.54	0.84	0.95	0.79	0.91	1	0.84	0.95	-0.23
average_viewers	0.72	1	0.88	0.88	0.86	0.84	1	0.88	-0.05
average_channels	0.54	0.87	1	0.78	0.92	0.95	0.88	1	-0.23
average_viewer_ratio	-0.074	-0.049	-0.23	-0.011	-0.17	-0.23	-0.05	-0.23	1

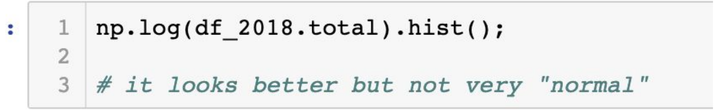
Appendix: Monthly Model





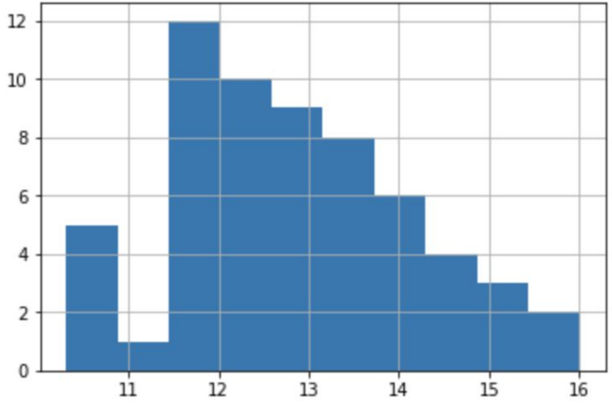
Appendix: Target Transformation

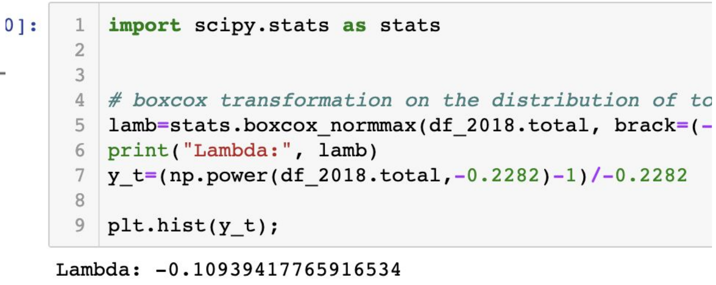




```
np.log(df_2018.total).hist();

# it looks better but not very "normal"
```



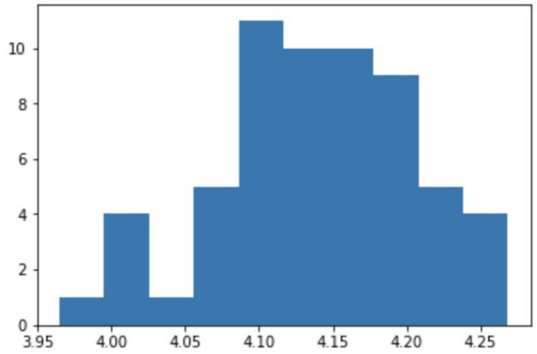


```
import scipy.stats as stats


# boxcox transformation on the distribution of to
lamb=stats.boxcox_normmax(df_2018.total, brack=(-
print("Lambda:", lamb)
y_t=(np.power(df_2018.total,-0.2282)-1)/-0.2282

plt.hist(y_t);
```

```
Lambda: -0.10939417765916534
```



10
8
6
4
2
0
3.95
4.00
4.05
4.10
4.15
4.20
4.25