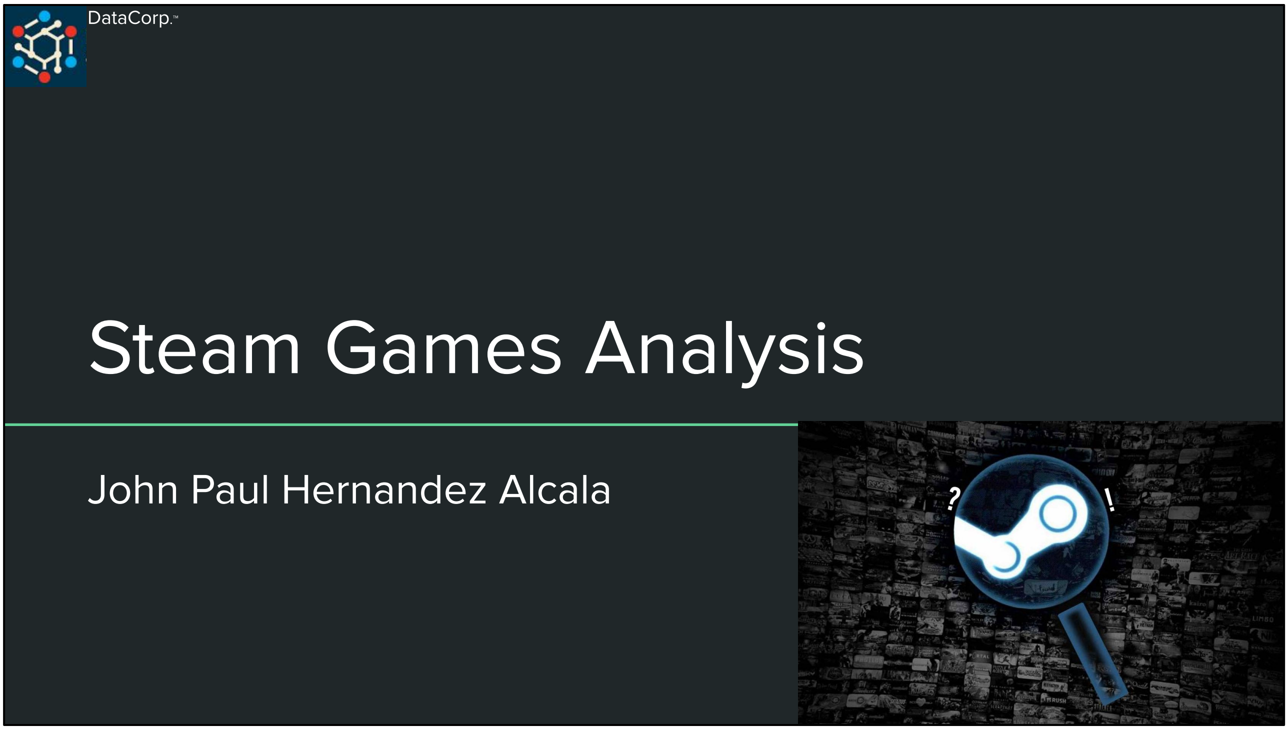


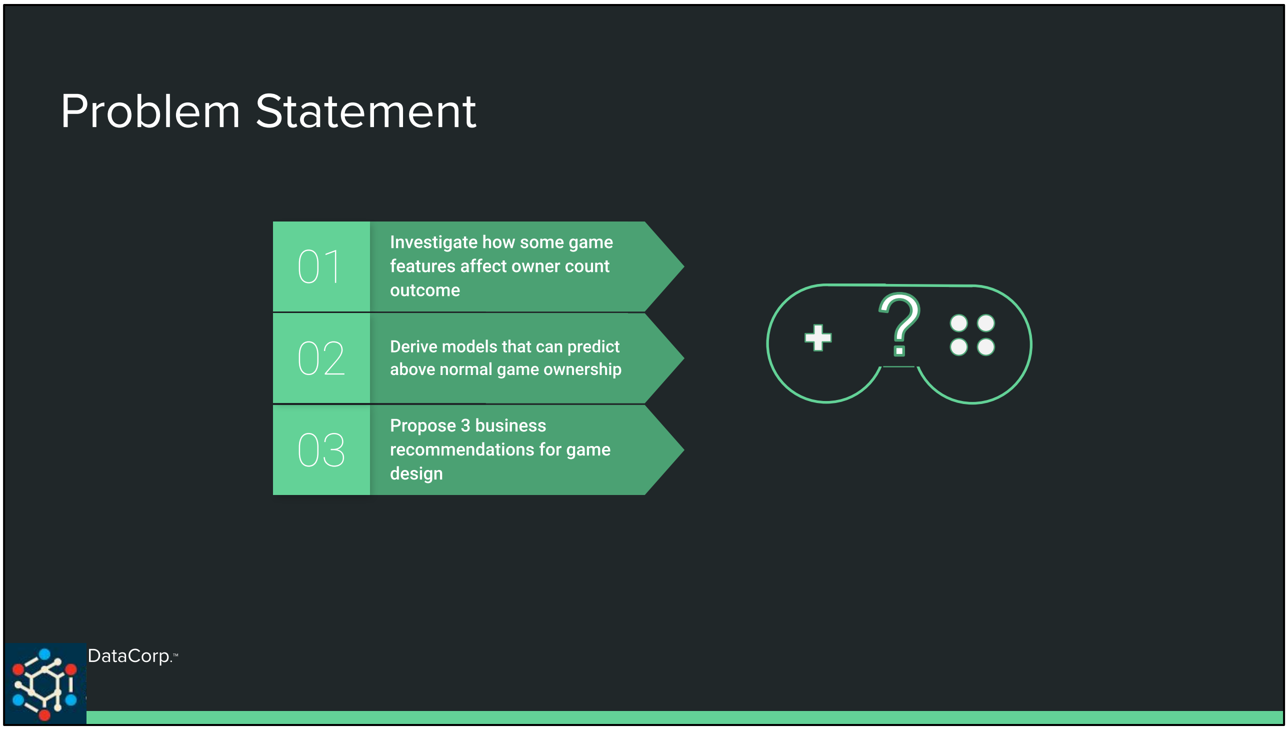
DataCorp.™

Steam Games Analysis

John Paul Hernandez Alcala

Hello, my name is John Paul, and I am a data scientist with DataCorp. Today my goal is to give you a better understanding of which games are more likely to be owned by Steam users. Let’s get started!

Image:
https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.slashgear.com%2Fsteams-customer-service-has-been-garbage-for-far-too-long-26604464%2F&psig=AOvVaw242iBdYFvAw4w2yaZJJsIW&ust=1610136130742000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCPClmrDYiu4CFQAAAAAdAAAAABAD



Problem Statement

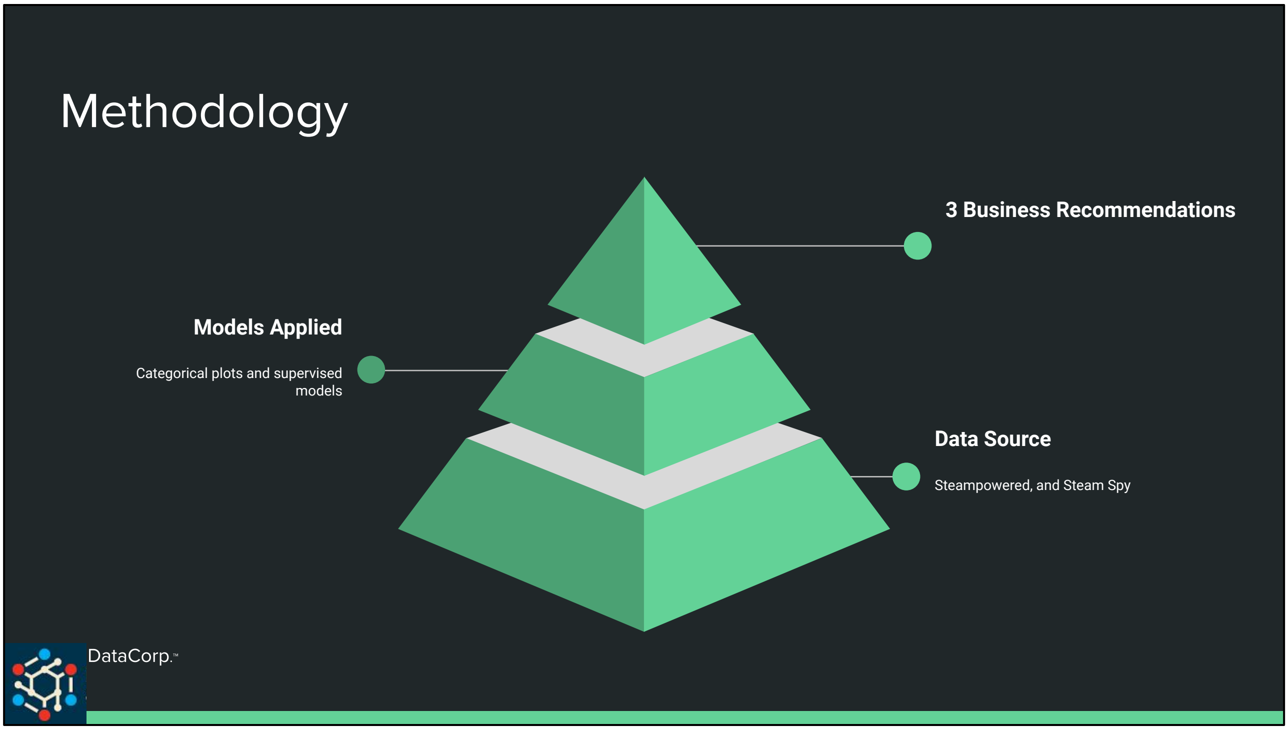
01 Investigate how some game features affect owner count outcome

02 Derive models that can predict above normal game ownership

03 Propose 3 business recommendations for game design

Before we dive into this, we have to present ourselves with three tasks that make up our problem statement: Investigate which game features affect owner number outcome, derive models that can predict games with above normal ownership, and propose 3 business recommendations for designing games with the most ownership

.



Methodology

3 Business Recommendations

Models Applied

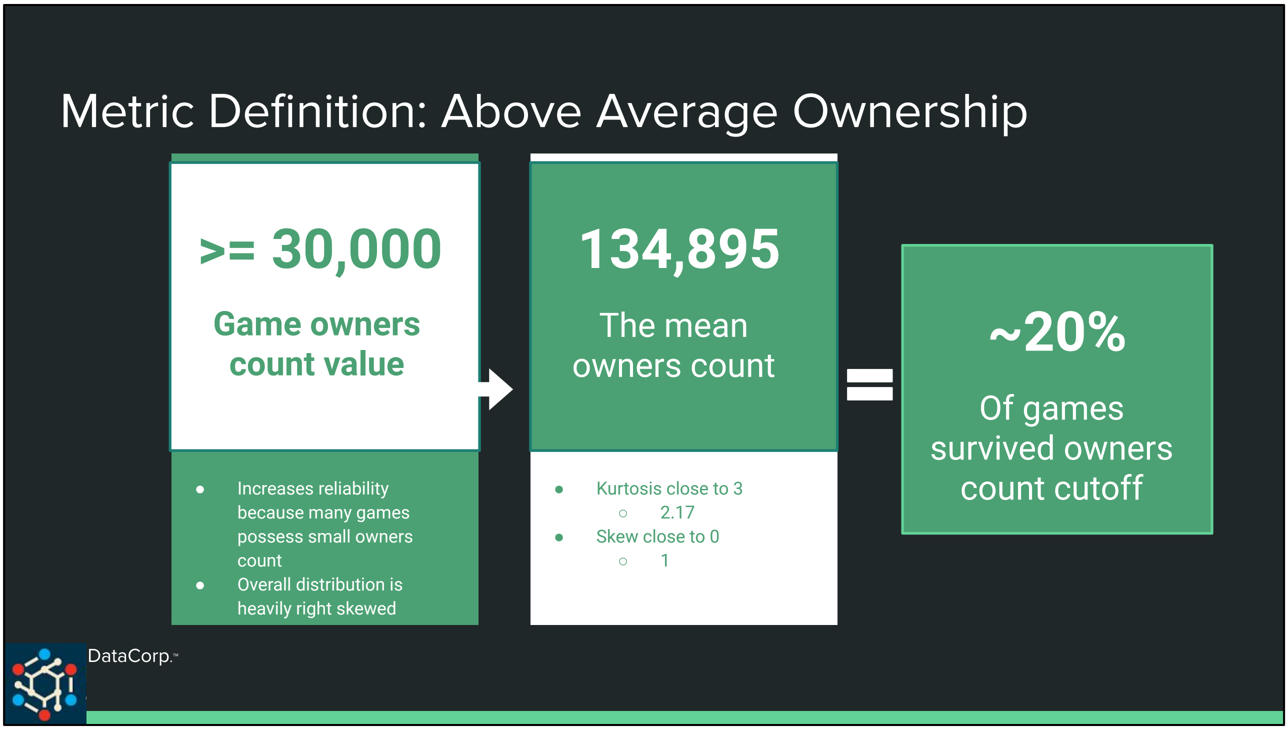
Categorical plots and supervised models

Data Source

Steampowered, and Steam Spy

DataCorp.™

To start on our journey of completing our tasks, we first start off by collecting data from both Steampowered and Steam Spy, work our way up with statistical models such as distribution plots, bar graphs, and point plots for categorical plots and decision tree and more ahead for supervised models. Finally, we will arrive to our three business recommendations.



Metric Definition: Above Average Ownership

>= 30,000

Game owners count value

- Increases reliability because many games possess small owners count
- Overall distribution is heavily right skewed

134,895

The mean owners count

- Kurtosis close to 3
  - 2.17
- Skew close to 0
  - 1

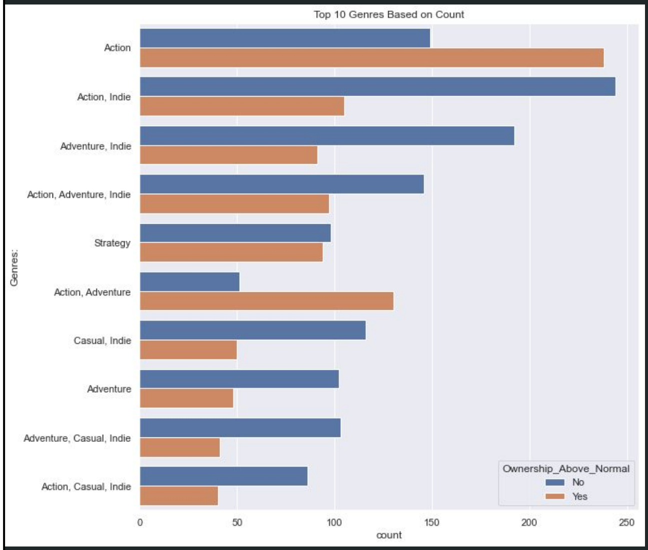
~20%

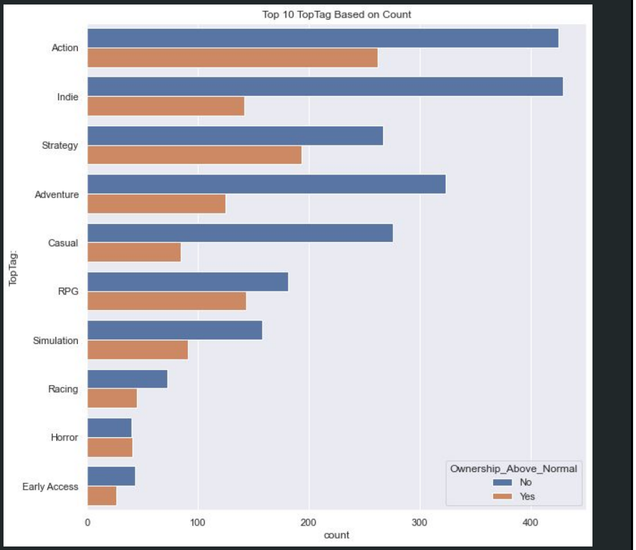
Of games survived owners count cutoff

DataCorp.™

Here we show how we established our owners cutoff

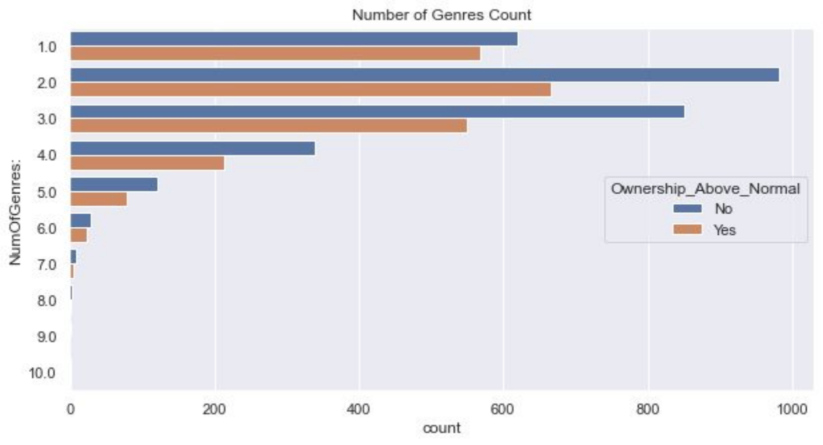
Bar Graph of ‘Genres’, and ‘TopTag’ Bar Graphs





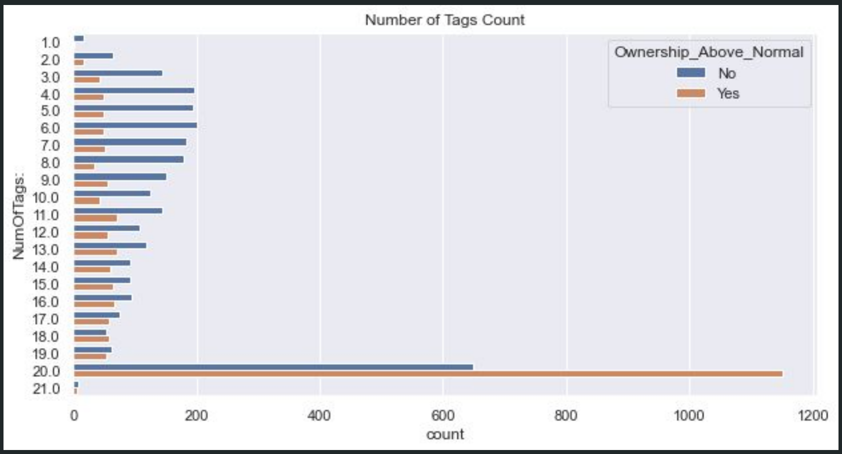
DataCorp.™

Bar Graph of ‘Genres’, and ‘TopTag’ Bar Graphs



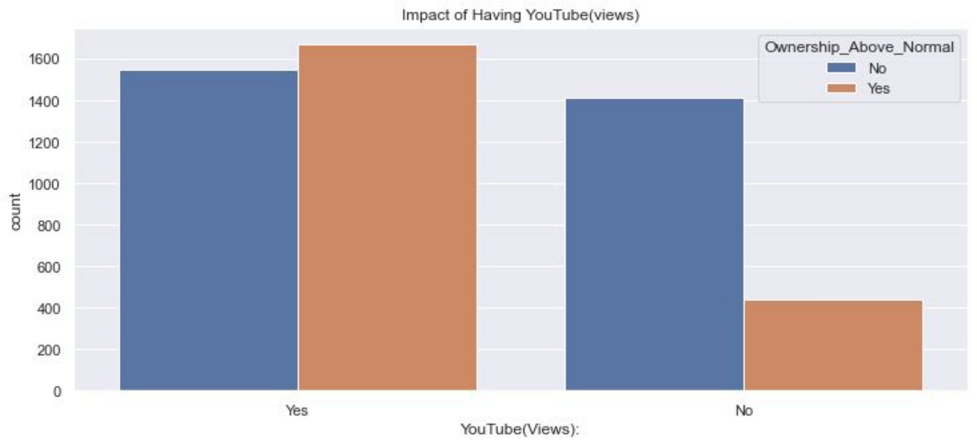
DataCorp.™

Bar Graph of ‘Genres’, and ‘TopTag’ Bar Graphs

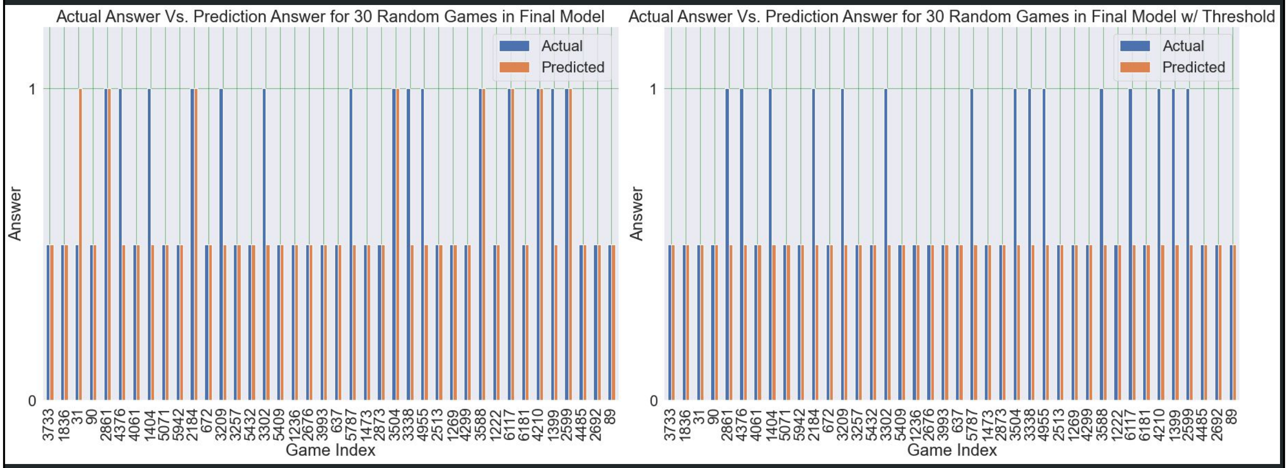


DataCorp.™

Bar Graph of ‘YouTube(views)’



Final Model vs Final Model with Threshold



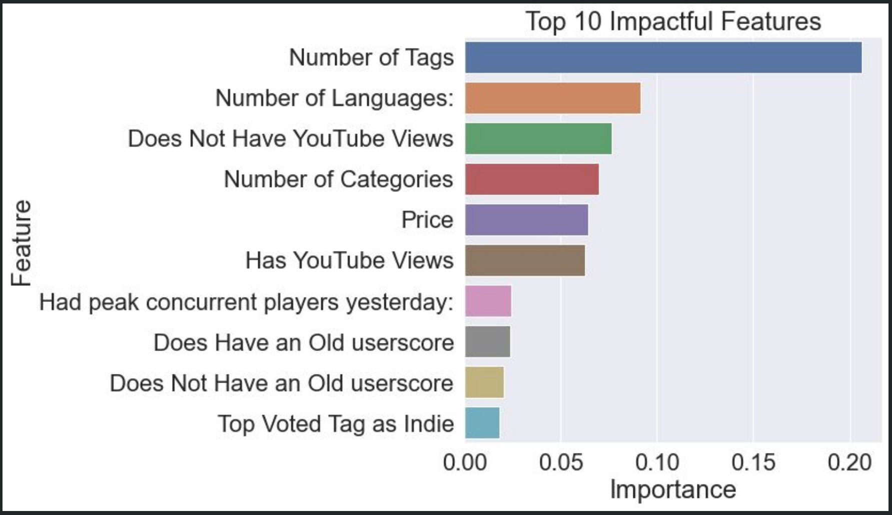
Actual Answer Vs. Prediction Answer for 30 Random Games in Final Model

Actual Answer Vs. Prediction Answer for 30 Random Games in Final Model w/ Threshold

DataCorp.™

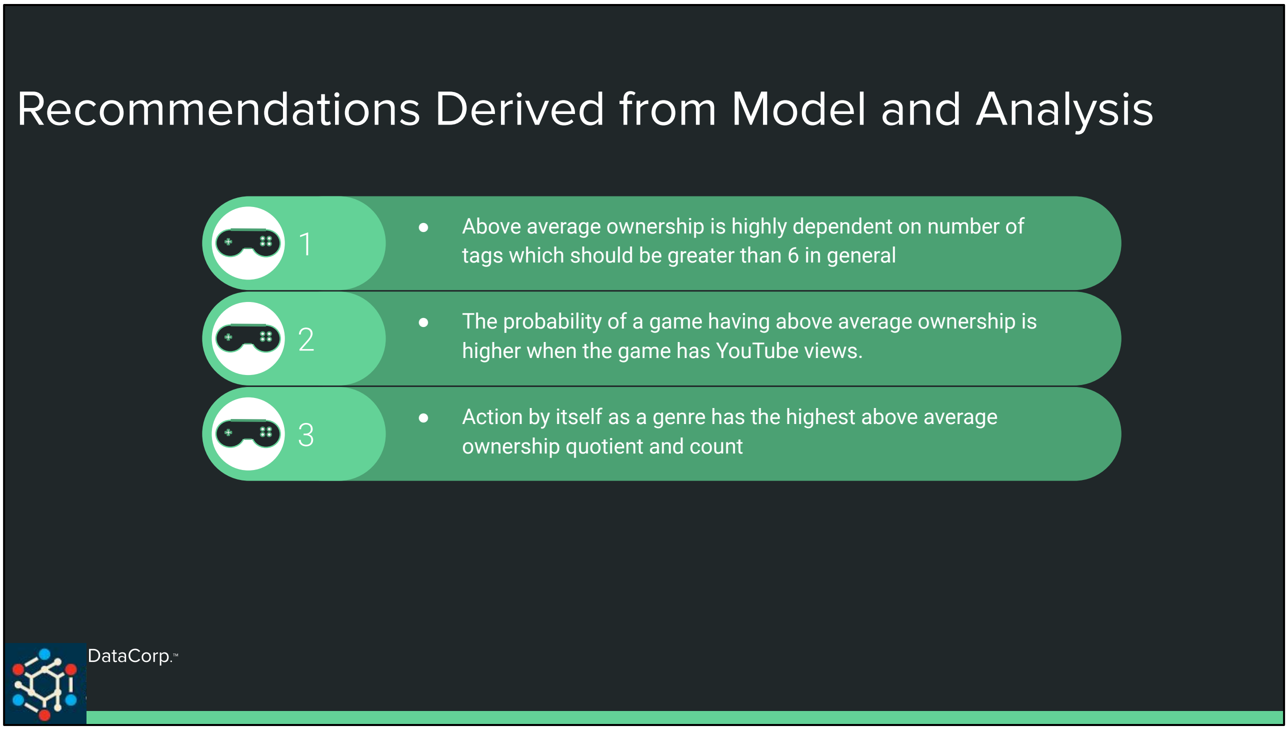
Here we see our model with and without a threshold implemented. This threshold makes our model more sensitive, but less specific and precise than the default model; that is, our threshold model allows for more false positives in order to reduce false negatives.

Top 10 Features



DataCorp.™

Here we see our model's top 10 features it employed.



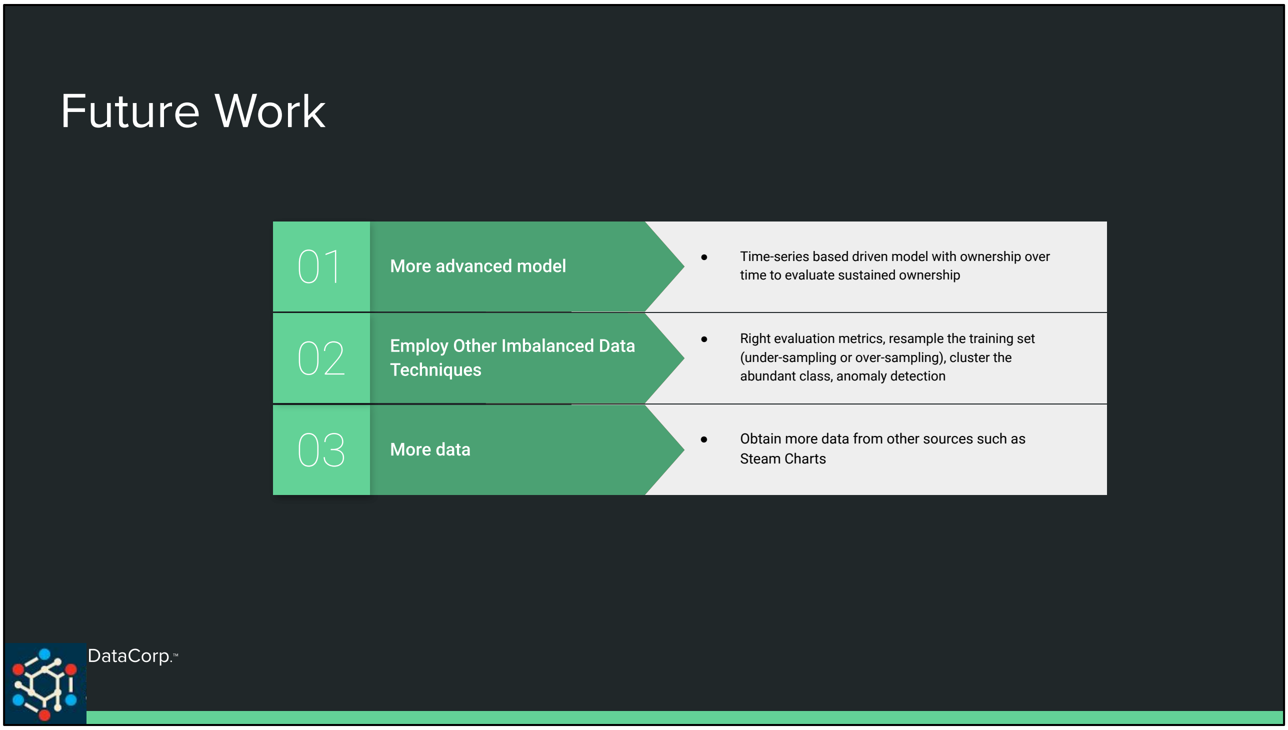
Recommendations Derived from Model and Analysis

1
- Above average ownership is highly dependent on number of tags which should be greater than 6 in general

2
- The probability of a game having above average ownership is higher when the game has YouTube views.

3
- Action by itself as a genre has the highest above average ownership quotient and count

DataCorp.™

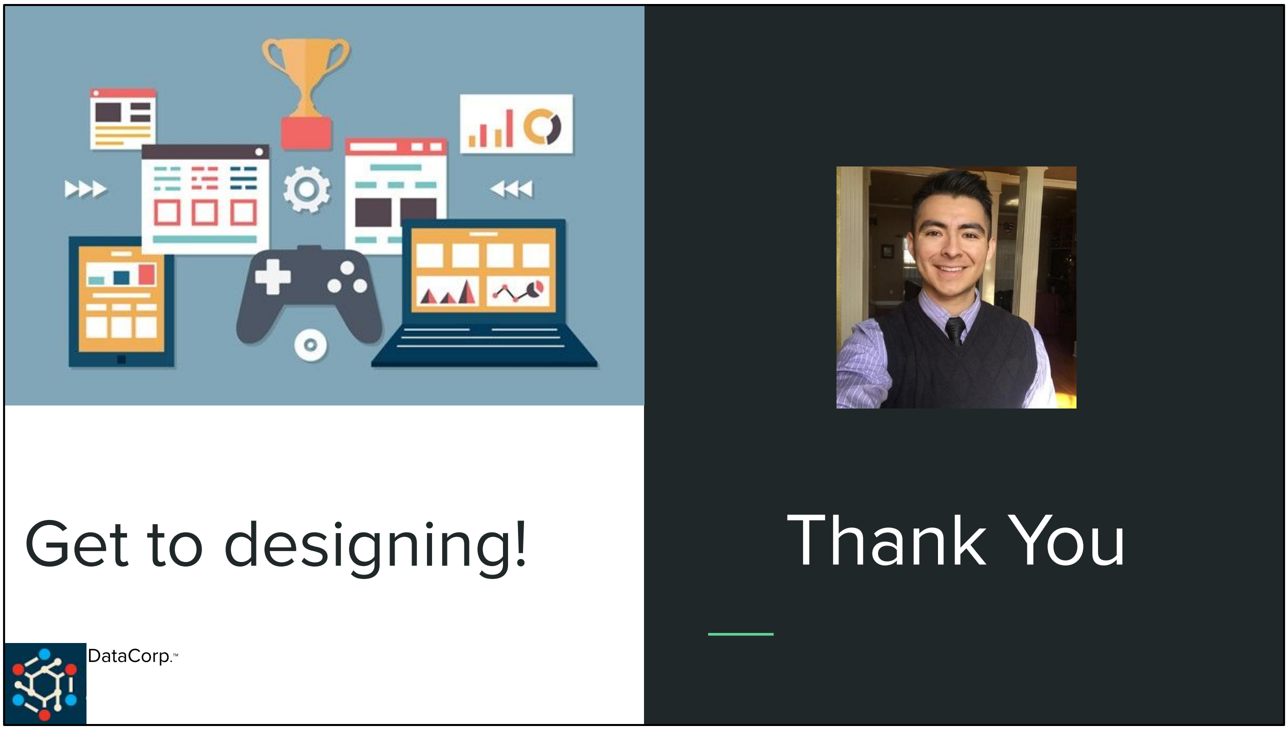


Future Work

01	More advanced model	• Time-series based driven model with ownership over time to evaluate sustained ownership
02	Employ Other Imbalanced Data Techniques	• Right evaluation metrics, resample the training set (under-sampling or over-sampling), cluster the abundant class, anomaly detection
03	More data	• Obtain more data from other sources such as Steam Charts

DataCorp.™

Now that we have proposed our three business recommendations, we know there are ways we can have more confidence with our predictors. We can use K Nearest Neighbor Imputation and Label Encoding instead of median and mode. We can use other imbalanced data techniques to improve recall, f1 score, and AUC. Finally, we can get more data to train and test against our current model which will allows us to work towards a better model.



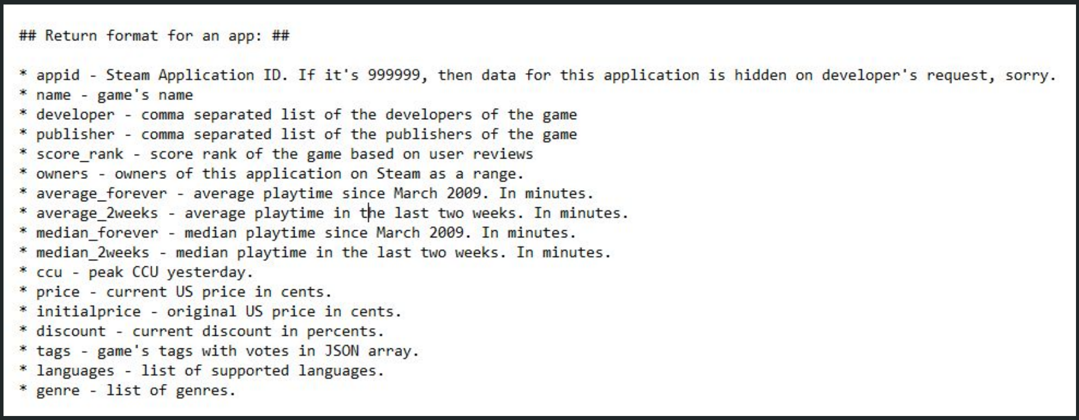
Get to designing!

DataCorp.™

Thank You

Thank you for your time! Please, feel free to ask me any questions at this time.

Appendix 1a. Features from Dataset



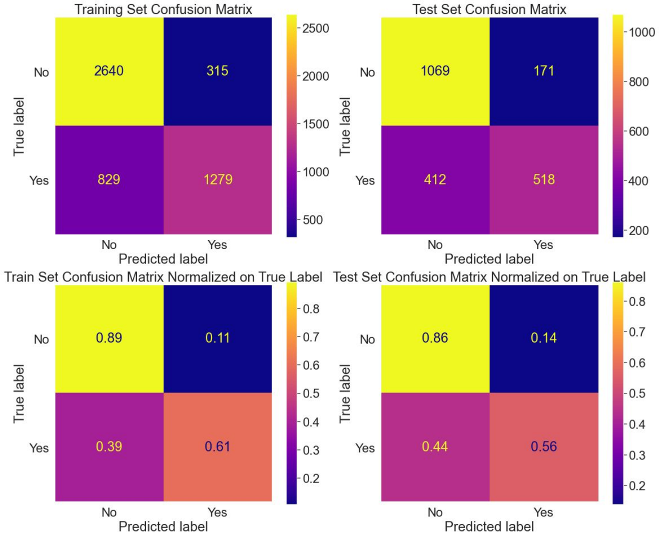
```
## Return format for an app: ##

* appid - Steam Application ID. If it's 999999, then data for this application is hidden on developer's request, sorry.
* name - game's name
* developer - comma separated list of the developers of the game
* publisher - comma separated list of the publishers of the game
* score_rank - score rank of the game based on user reviews
* owners - owners of this application on Steam as a range.
* average_forever - average playtime since March 2009. In minutes.
* average_2weeks - average playtime in the last two weeks. In minutes.
* median_forever - median playtime since March 2009. In minutes.
* median_2weeks - median playtime in the last two weeks. In minutes.
* ccu - peak CCU yesterday.
* price - current US price in cents.
* initialprice - original US price in cents.
* discount - current discount in percents.
* tags - game's tags with votes in JSON array.
* languages - list of supported languages.
* genre - list of genres.
```

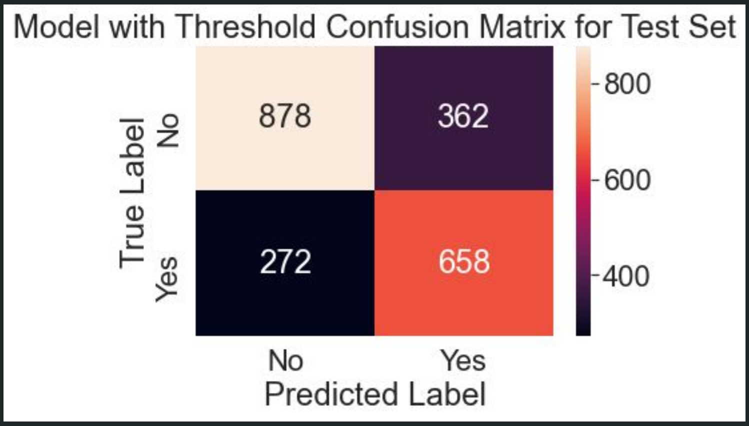
DataCorp™

Key for features

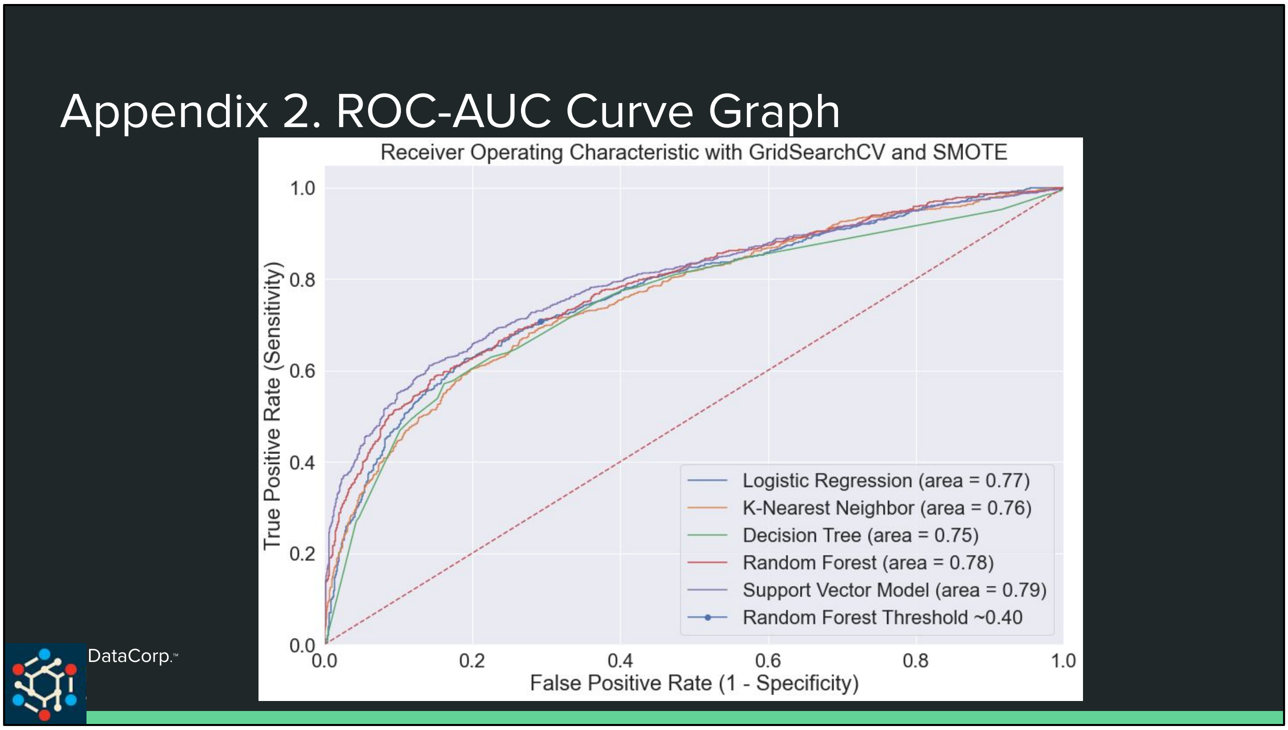
Appendix 1b. Confusion Matrix for Final Model



Appendix 1b. Confusion Matrix for Final Model with custom threshold



DataCorp.™



Appendix 2. ROC-AUC Curve Graph

From the above, we see that the best model to focus on is the random forest one since it resulted in a AUC of 0.78 in our ROC Curve Graph. We have talked about F1 and recall score, but we have not discussed what AUC means for our model. Generally, the higher the AUC, better the model is at distinguishing between games that will have above average ownership outcome and will not. So an AUC of 0.78 means that there is 78% chance that the model will be able to distinguish between 'yes' class and 'no' class.