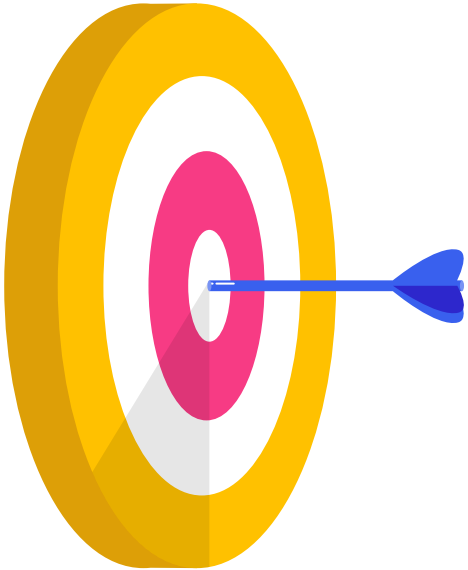
Video Game Recommendation Tool

Using Google Colab, Machine Learning, Flask API, SQL, HTML/CSS, JavaScript, & Python Pandas

Alvaro, Dozie, Emmanuel, Erick, Jonathan, & Mickey



Goal 01.

Build on existing infrastructure & past successes. Create a one-stop-shop for entertainment solution

Goal 02.

Position Group 1 as sought after consultants & subject-matter experts in the video game industry, for both B2C and B2B

Goal 03.

Create a tool that is personable, relatable, scalable, & capable

Goal 04.

Leverage data analytics & data science to create actionable insights

Our Process

Selecting the "Idea"

- Predict Sales & Scores
- Pivot to new ideas to suit our data set
- Select best idea that works before writing code

Writing the Code

- Review project requirements
- Learn new libraries and packages such as NLTK and Scikit-Learn

Crafting User Experience

- Identifying limitations
- Drop-down vs. user-input
- Error message experiences
- Designing look & feel
- Mock ups
- Branding

HTML Implementation

- Integration with existing infrastructure
- Building form fields
- Connecting Python & HTML
- Writing functions

Preview of Recommendation Tool

Specific Games

Get recommendations based on your favorite game

Developers/Publishers

Get recommendations based on your favorite Developers & Publishers

Year/Rating

Get recommendations based on the year or rating

Sentences/Prompts

Get recommendations based on search queries or prompts

How it works

Text-based queries

Enter a query into our website

Cross-referencing 4,000+ games

The model takes that game and runs it through a similarity algorithm and assigns each game in the database a numerical value or “score”

User Input → **Best Match** → **Similarity Ranking** → **Recommendation Engine**

Identifying the best match

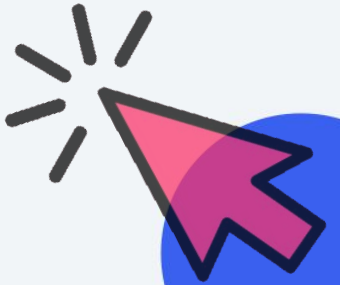
Our model tries to find the one game that is most similar to your input

Top recommendations

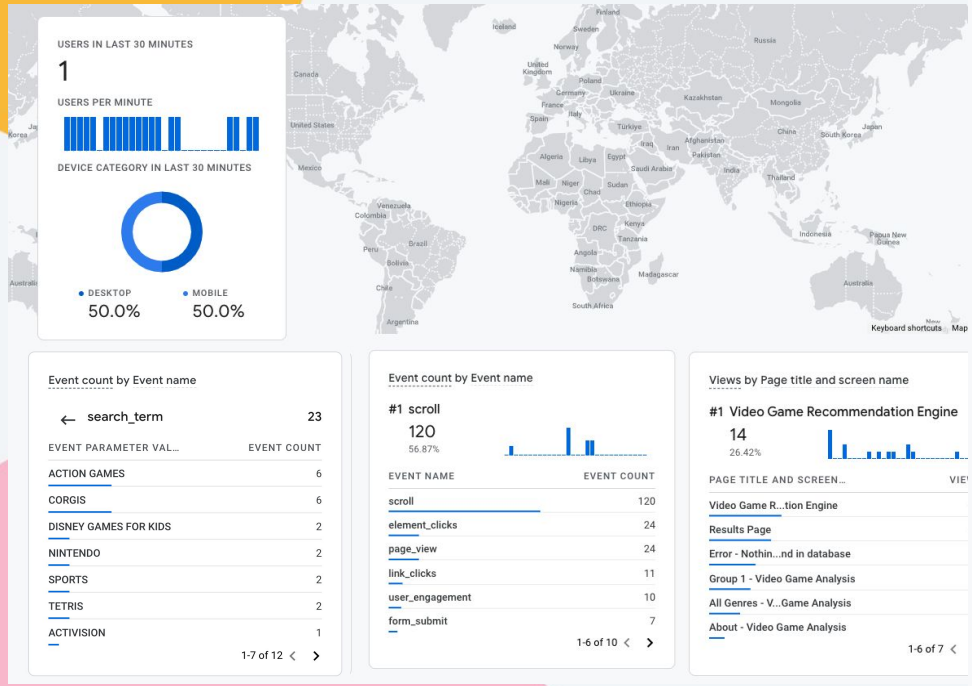
The games with the highest scores are presented as your personal recommendations

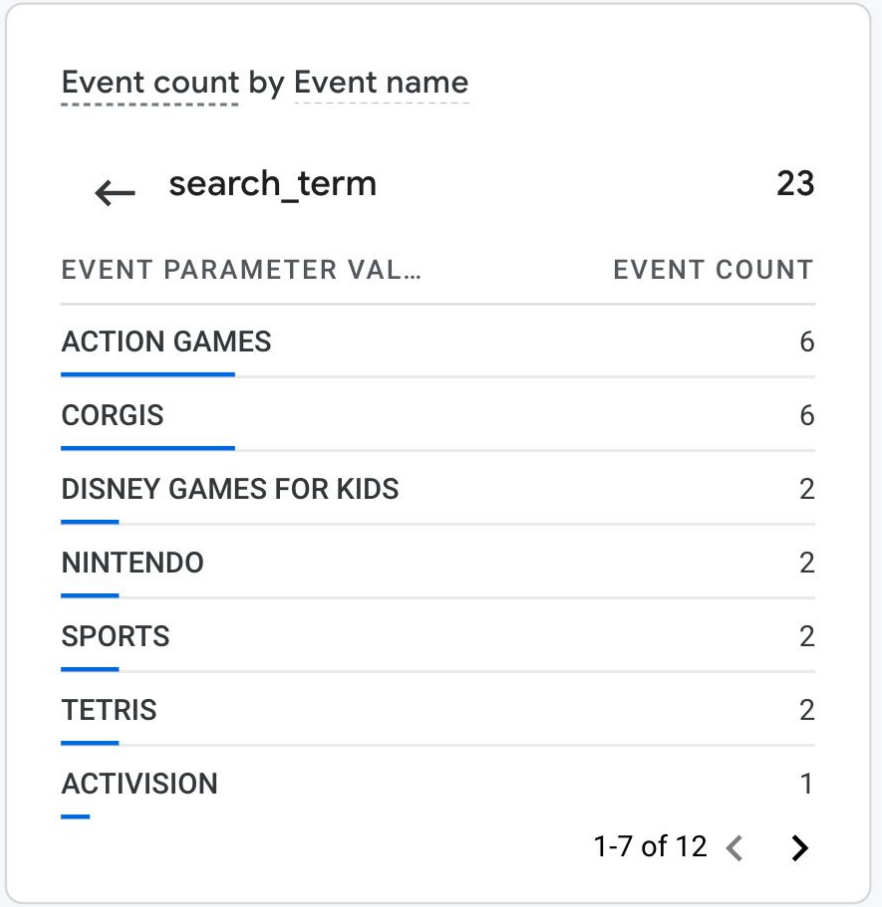
Live Demo

Video Game Recommendation Engine



GOOGLE ANALYTICS || PREVIEW





VIDEO GAME DASHBOARD | Website Performance [localhost]

Jun 7, 2023 - Jun 8, 2023

SESSIONS	PAGEVIEWS	ELEMENT CLICKS	LINK CLICKS	SEARCHES
16	417	671	207	135
⬆ 1,500.0%	⬆ 2,094.7%	⬆ 11,083.3%	-	⬆ 4,400.0%

Sessions

Clicks

Game Queries / Searches

	Page title	Views
1.	Video Game Recommendation Engine	148
2.	Results Page	110
3.	All Genres - Video Game Analysis	41
4.	Error - Nothing found in database	32
5.	Action - Video Game Analysis	28

1 - 14 / 14

	Link Click Text	Link Click URL	Clicks
1.	Recommendation Tool	http://127.0.0.1:5000/recomme...	56
2.	Return to search	http://127.0.0.1:5000/recomme...	35
3.	Action	http://127.0.0.1:5000/actiondas...	27
4.	Dashboard	http://127.0.0.1:5000/	26
5.	Team	http://127.0.0.1:5000/team	16

1 - 20 / 20

	Search Term	Searches
1.	CORGIS	21
2.	DIABLO	15
3.	I JUST WATCHED THE MARI...	8
4.	ACTION GAMES	6
5.	NINTENDO	6

1 - 65 / 65

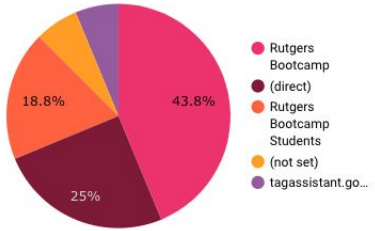
Top Campaigns / Traffic Sources

	Campaign	Source	Medium	Sessions
1.	Group 1 Presentation	Rutgers Bootcamp	Cohort Link	7
2.	(direct)	(direct)	(none)	4
3.	(not set)	Rutgers Bootcamp Students	(not set)	3
4.	(not set)	(not set)	(not set)	1

1 - 5 / 5

User Location

Source Breakout



Recommendation Tool Findings

We Like the Recommendations

- Suits different preferences
- Seems accurate across a wide breadth of queries
- We found it hard to disagree with the recommendations

The Dreaded “Zumba” Effect

- No results occurs too often, ultimately recommending the Zumba Fitness Rush & similar games
- Issue fixed, but highlights limitations of dataset

Room for Adjustments

- Showing older iterations of the same game (i.e. Madden 09)
- Dataset doesn’t have some of the most popular games

images via Clipart, Gyfcat Library, & YouTube

Project Limitations & Future Enhancements

Predictive Modeling Ain't Easy

- Developed multiple models with low accuracy/R2 scores
- Find new datasets that better fit our models
- Incorporate web analytics into models

Limited Dataset

- Only up to 2016
- Missing many games
- No platform/consoles
- A simple model due to lack of features

Enhance the User Experience

- Include video game cover images, links to info/purchase
- Create more personalized recommendations
- Mobile friendly, voice search

Optimize Coding Approaches

- Adjust HTML & CSS
- Condense Flask & SQL queries
- Enhance Google Analytics
- Search is still buggy

ML Classification Models

globalsales	g_sales	criticscore	criticcount	userscore	usercount
1.27	1	75	35	8.5	60
0.68	0	76	24	8.9	81
0.46	0	70	23	8.7	19
1.31	1	78	83	7.8	356
1.24	1	79	53	7.7	308

```
#Logistic Regression
target_names = ['low', 'high']
predictions = model.predict(X_test)
print(classification_report(y_test, predictions,
                            target_names=target_names))
```

```
              precision    recall  f1-score   support

         low       0.82      0.96      0.88      1306
        high       0.72      0.34      0.46       418

    accuracy                           0.81      1724
   macro avg       0.77      0.65      0.67      1724
weighted avg       0.79      0.81      0.78      1724
```

```
#Decision Tree Model
predictions = model_3.predict(X_test)
print(classification_report(y_test, predictions,
                            target_names=target_names))
```

```
              precision    recall  f1-score   support

         low       0.81      0.82      0.81      1306
        high       0.42      0.41      0.41       418

    accuracy                           0.72      1724
   macro avg       0.62      0.61      0.61      1724
weighted avg       0.72      0.72      0.72      1724
```

```
#Support Vector Machine
target_names = ['low', 'high']
predictions = model_2.predict(X_test)
print(classification_report(y_test, predictions,
                            target_names=target_names))
```

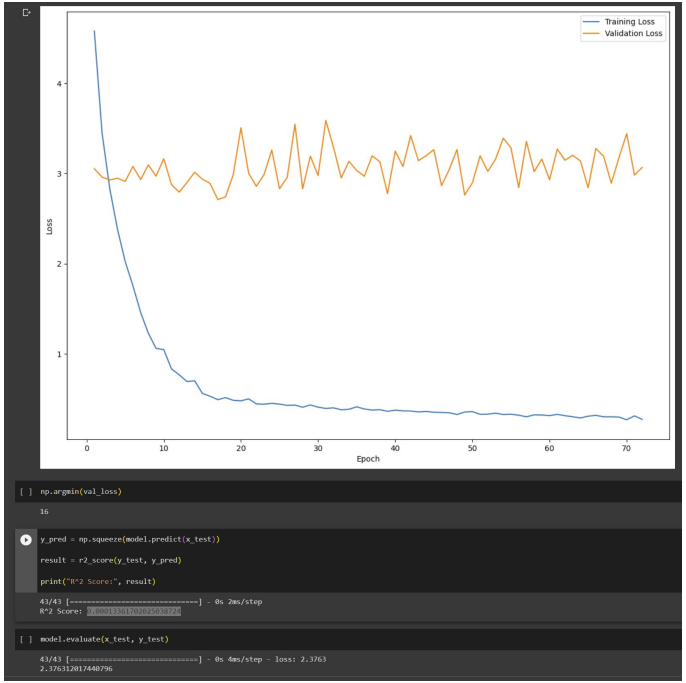
```
              precision    recall  f1-score   support

         low       0.79      0.96      0.87      1306
        high       0.61      0.19      0.29       418

    accuracy                           0.77      1724
   macro avg       0.70      0.57      0.58      1724
weighted avg       0.75      0.77      0.73      1724
```

Final Thoughts on Sales Prediction Model

- The sales prediction model had a low R^2 score (0.00013361702025038724)
- This is likely due to the fact that the column data was based on game name, year released, genre, publisher, rating, global sales, critic score, critic count, user score, and user count.
- None of this data seems to correlate to each other.
- We believe in the future, collecting additional data, such as number of copies sold or price per copy, we would see a stronger correlation.



Similarity Search Accuracy

- Snapshot of code and similarity scores from a search

	0	1	2	3	4	5	6	7	8	9
0	1.000000	0.111803	0.144338	0.125000	0.000000	0.000000	0.000000	0.111803	0.000000	0.000000
1	0.111803	1.000000	0.000000	0.111803	0.000000	0.000000	0.119523	0.000000	0.000000	0.000000
2	0.144338	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.258199	0.000000	0.000000
3	0.125000	0.111803	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
4	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.192450	0.353553
...	...	...	...	...	...	...	...	...	...	...
4415	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.100000	0.182574	0.000000
4416	0.000000	0.000000	0.000000	0.000000	0.117851	0.000000	0.000000	0.000000	0.102062	0.125000
4417	0.133631	0.119523	0.000000	0.133631	0.000000	0.142857	0.000000	0.000000	0.000000	0.000000
4418	0.000000	0.000000	0.000000	0.000000	0.000000	0.239046	0.000000	0.000000	0.000000	0.000000
4419	0.000000	0.000000	0.000000	0.000000	0.117851	0.000000	0.000000	0.111803	0.204124	0.125000

4420 rows × 4420 columns

THANKS!

Questions?

Alvaro, Dozie, Emmanuel, Erick, Jonathan, & Mickey

Who wants to volunteer?!

Question for the class:

*Give us a **query** to run in our recommendation tool!*