Increase Revenue For Catch The Pink Flamingo Game

NOV. 28/2022

KEDIR NASIR OMER

Content

- ✓ Project Background
- ✓ Data Exploration Overview
- ✓ What have we learned from classification?
- ✓ What have we learned from clustering?
- ✓ From our chat graph analysis, what further exploration should we undertake?
- ✓ Recommendation
- ✓ Reference
- ✓ Acknowledgment

Project
Background

➢ Project's Aim

- ✓ **Unleash** the power of data and analytics to **increase revenue**, find **new business** opportunities and learn more about player behavior.

➢ Problems I'm going to solve

- ✓ Promoting the game to draw more users from **iOS and Mac**. High Rollers who are willing to spend more are more likely to play on these two platforms.
- ✓ Concentrate on marketing and creating in-app purchase items like **itemID 5**, which is the most lucrative one.
- ✓ By **promoting** the most influential **players**, they could quickly draw in the neighborhood.

➢ Data Source

✓ Players activities during the game

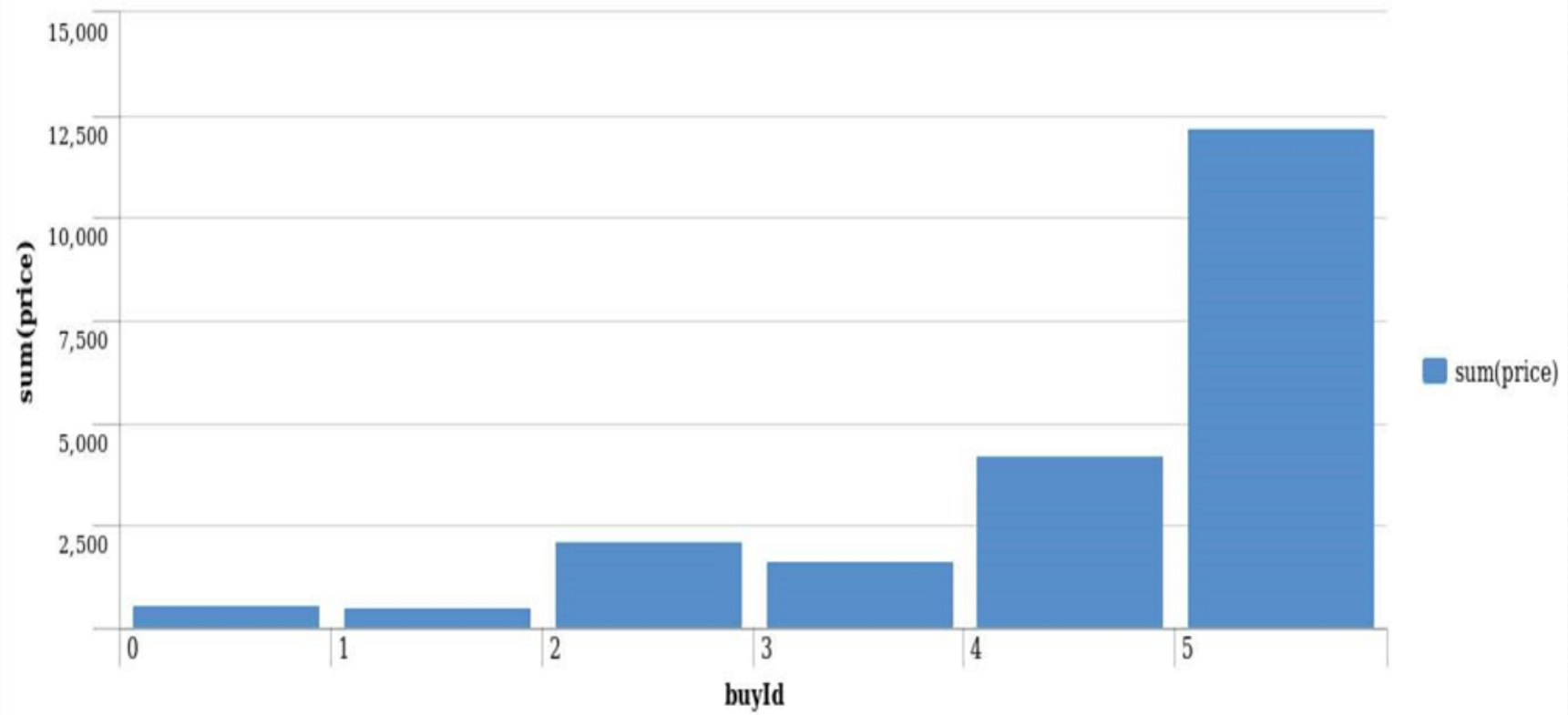
This data enables us to assess when and where users click on the screen, when they make an in-app purchase, and when they click on a banner. Understanding this data enables us to assess and create more user-friendly game experiences that convert, target promotions to certain users, and develop pricing strategies.

✓ Chats between players

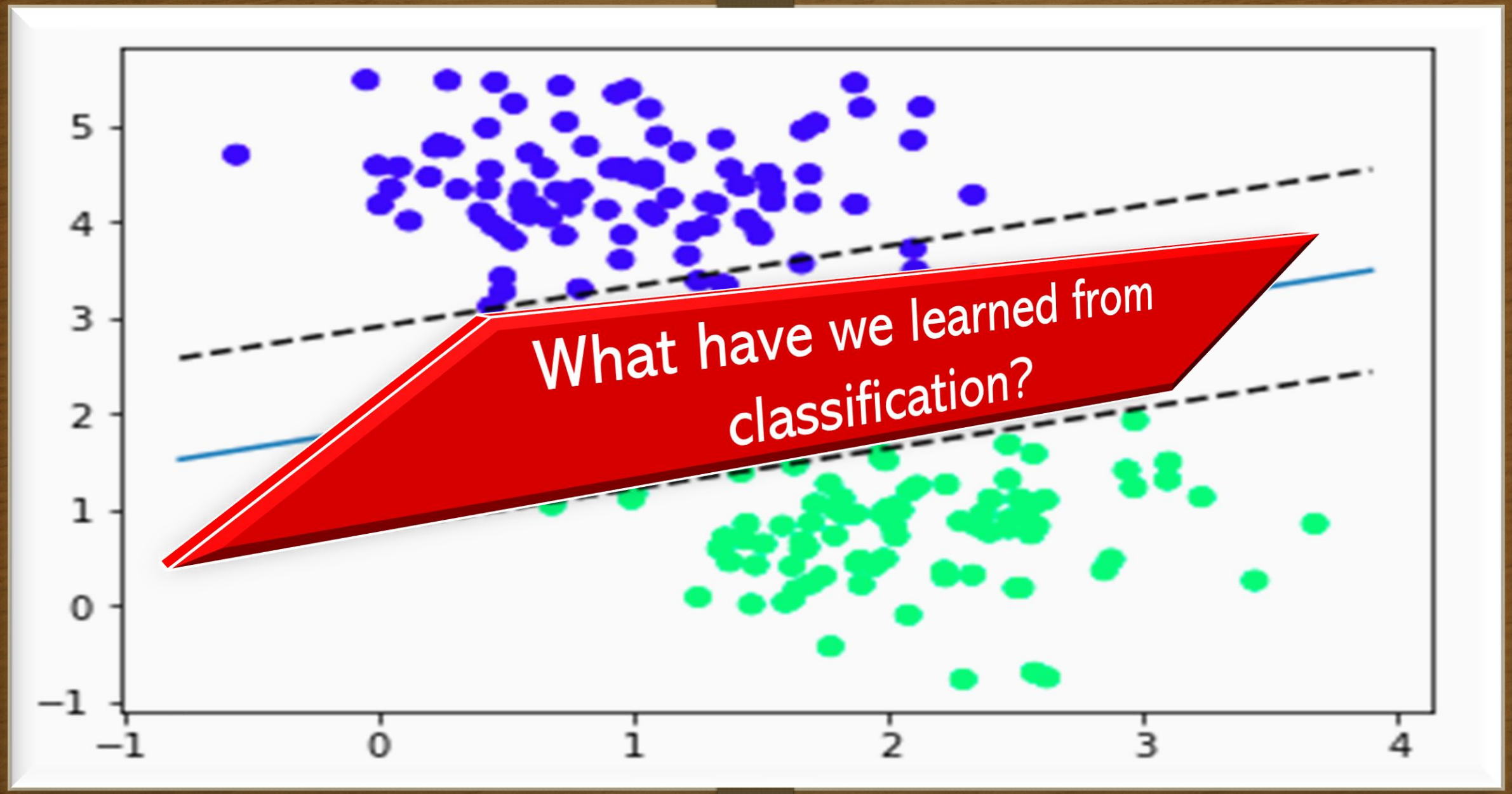
We can identify the most important players and the hottest subjects using this data, which is very helpful. Based on the information we learned from this data, we can create marketing campaign strategies that are more effective.

Data Exploration Overview

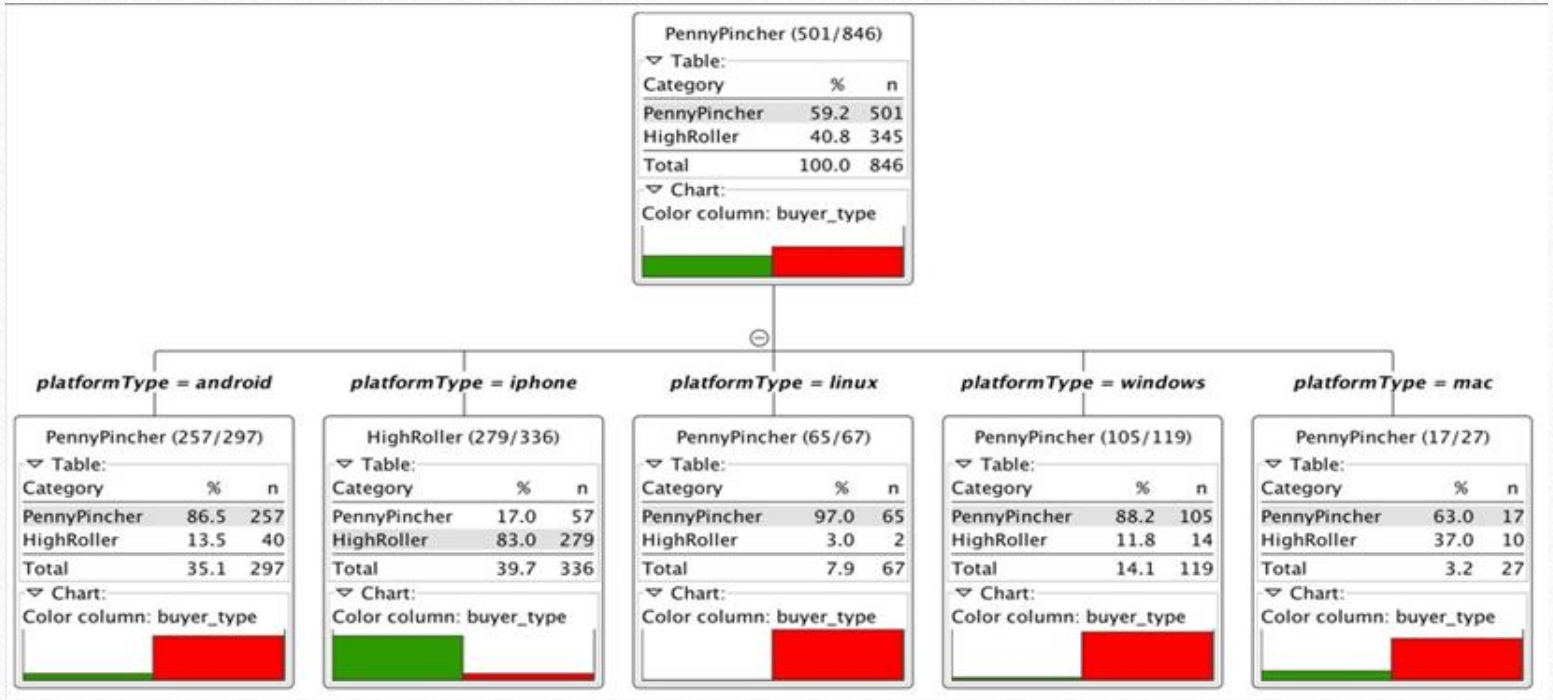
- Revenue generated from In-app purchase items



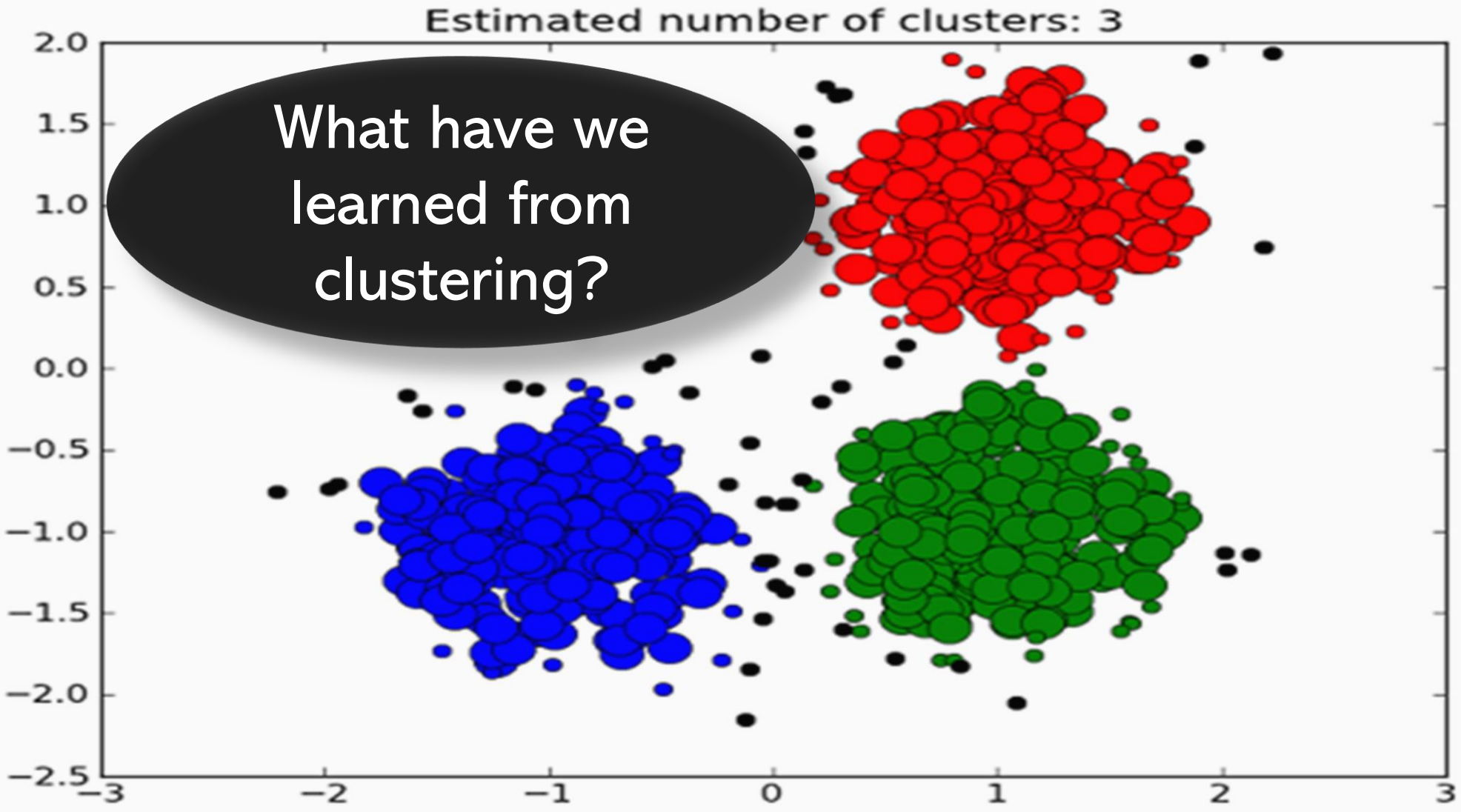
More money is made from ItemID 5 than all other items combined by 57%. We could create a marketing plan that encourages itemID sales.


What have we learned from classification?
5
4
3
2
1
0
−1
−1
0
1
2
3
4

➢ Put an emphasis on iPhone users



Most players are on mobile platforms, Iphone player are likely to be HighRoller while android players tend to be PennyPinchers. Promoting game among iOS and Mac users will increase the revenue.


Estimated number of clusters: 3
What have we learned from clustering?
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
−3
−2
−1
0
1
2
3

Cluster #	Cluster Center
1	[41.07, 10.29, 145.51]
2	[34.28, 6.45, 67.22]
3	[26.30, 4.48, 17.07]

Cluster centers

totalAdClinks: Total number of ad-clicks per user
totalBuyClicks: Total number of in-app purchase per user
totalRevenue: Total money spent on in-app purchase per user.

- These clusters can be distinguished from one another in the listed ways

- ✓ In contrast to the other clusters, **Cluster 1's** participants have the **highest** "totalAdClics," "totalBuyClics," and "totalRevenue."

- ✓ The players in **Cluster 2** differ from the others in that they have the **second-highest** "totalAdClics," "totalBuyClics," and "totalRevenue."

- ✓ The players in **Cluster 3** differ from the others in that they have the **lowest** "totalAdClics," "totalBuyClics," and "totalRevenue" numbers.

From our chat graph analysis, what further exploration should we undertake?

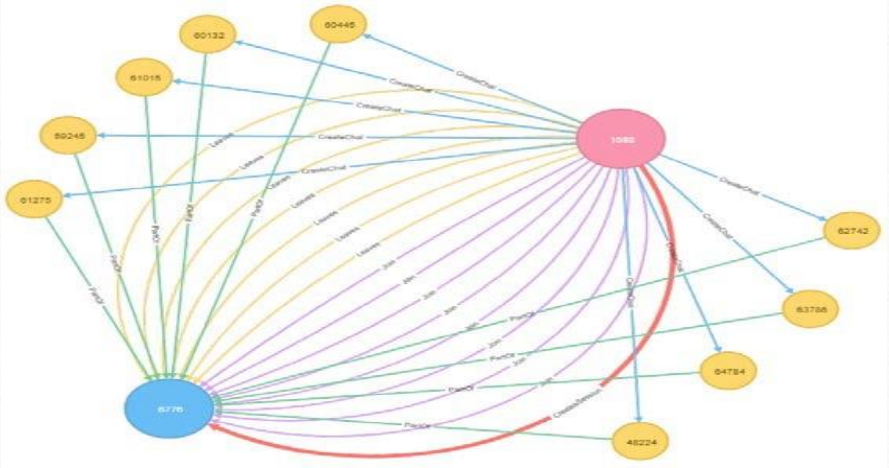
➢ Graph Analytics with Chat Data Using Neo4j

- ✓ Modeling Chat Data using a Graph Data Model

A network based on user chat interactions makes up the graph model. Users on the same team can join and leave a chat session, which can be started by one of them

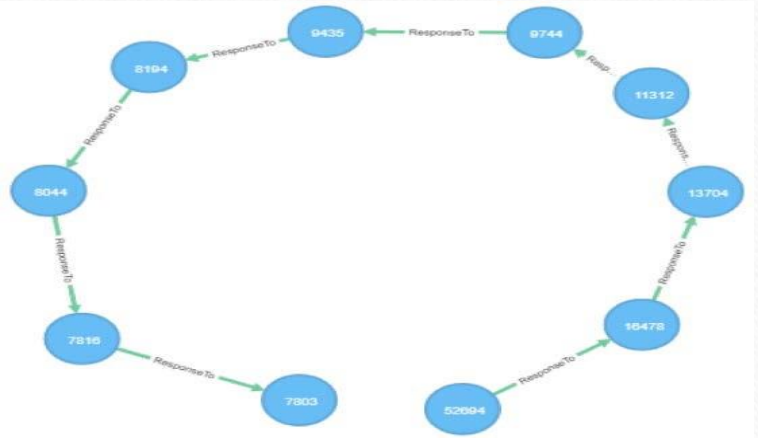
- ✓ Creation of the Graph Database for Chats

```
LOAD CSV FROM "file:///chat-data/chat_create_team_chat.csv" AS row
MERGE (u:User {id: toInteger(row[0])})
MERGE (t:Team {id: toInteger(row[1])})
MERGE (c:TeamChatSession {id: toInteger(row[2])})
MERGE (u)-[:CreatesSession{timeStamp: row[3]}]->(c)
MERGE (c)-[:OwnedBy{timeStamp: row[3]}]->(t)
```



- ✓ Finding the longest conversation chain and its participants

```
MATCH p=(a)-[:ResponseTo*]->(b)
RETURN p, length(p)
ORDER BY length(p) desc limit 1
```



- ✓ Analyzing the relationship between top 10 chattiest users and top 10 chattiest teams

```
match p=(c:ChatItem)-[:ResponseTo*]->(j:ChatItem)
where length(p)=9
with p
match q=(u:User)-[:CreateChat]-(c:ChatItem)
where (c IN NODES(p))
return count(distinct u)
```

```
match (u:User)-[:CreateChat]-(i:ChatItem)
return u.id as Users, count(u.id) as Num_Chats
order by count(u.id) desc limit 10
```

Users	Num_Chats
394	115
2067	111
209	109
1087	109
554	107
516	105
1627	105
999	105
668	104
461	104

Recommendation

➢ Solutions and Next steps

- ✓ Focus on selling and developing **in-app** purchase items like the **itemID 5**, it's the most profitable item
- ✓ **Promoting** the game to attract more **iOS and Mac users**. Players on these two platform are more likely to be High Roller who are willing to spend more
- ✓ Pushing **promotion** to the most influencer **players**, they could easily attract the community

Reference

- Big Data Specializations: Big Data Capstone Project

  - ✓ Visit My GitHub Portfolio: https://github.com/kedibeki/Increase-Revenue-For-Catch-The-Pink-Flamingo-Game
  - ✓ Visit UC SanDiego Coursera Page: https://www.coursera.org/learn/big-data-project?specialization=big-data

Acknowledgment

- UC SanDiego and Coursera: Big Data Specializations Instructors
  - ✓ Ilkay Altintas- Chief Data Science Officer
  - ✓ Amarnath Gupta- Director, Advanced Query Processing Lab
  - ✓ Mai Nguyen- Lead for Data Analytics
  - ✓ UC SanDiego and Coursera
  - ✓ Peer Learners

TARGET FOR SUCCESS!

Thank You!