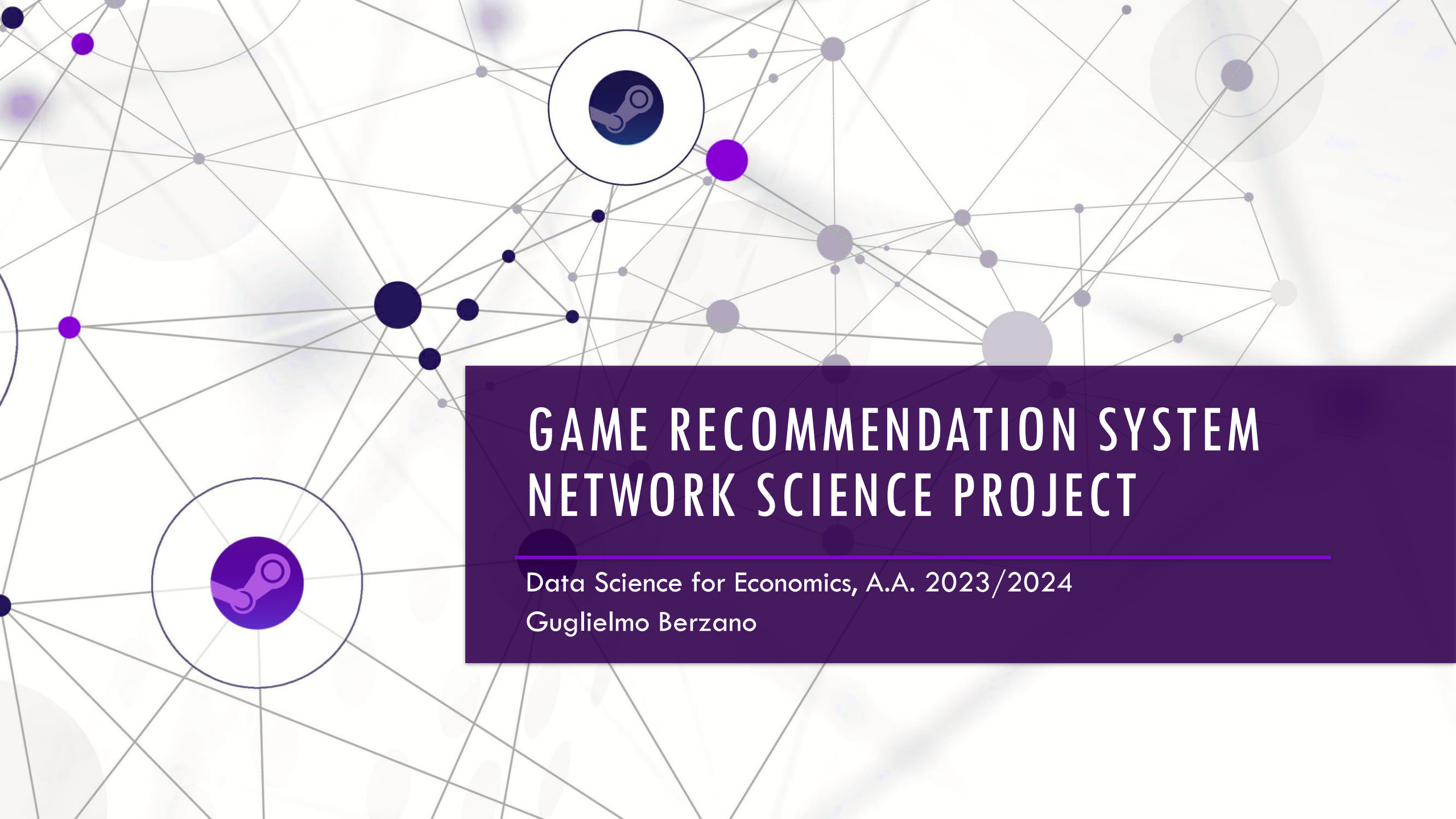


GAME RECOMMENDATION SYSTEM NETWORK SCIENCE PROJECT

Data Science for Economics, A.A. 2023/2024

Guglielmo Berzano

TABLE OF CONTENTS

INTRODUCTION

DATA EXPLORATION AND PREPROCESSING

GRAPH ANALYSIS

NODE2VEC ALGORITHM

CONCLUSIONS

INTRODUCTION – PROJECT'S AIM

The project's aim is to create a RecSys based on real user-reviews published on the Steam website from June 2022 to December 2022 by using the Node2Vec algorithm.

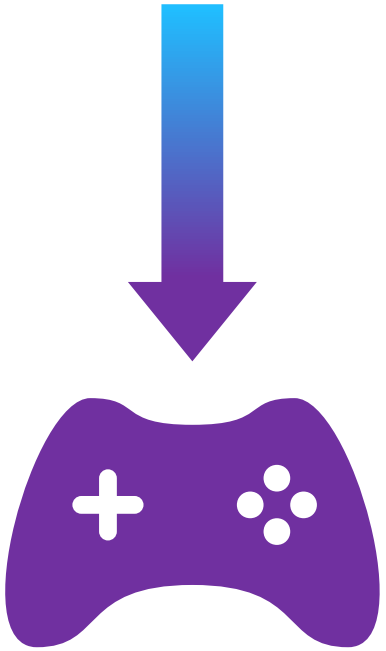
INTRODUCTION – WHAT IS STEAM?

Steam is one of the biggest video games distribution platforms with over 73 thousands games availabe, allowing developers worldwide to generate from 50 to 99 percent of their total income.

Source: Statista

INTRODUCTION

- Dataset taken from Kaggle
- 21 total columns, most relevant:
  - Game id
  - Game title
  - Game popularity
  - User id (anonymized)
  - Hours played
  - General rating
  - Recommandation



- 41.2 million rows, reviews spanning from October 2010 to December 2022

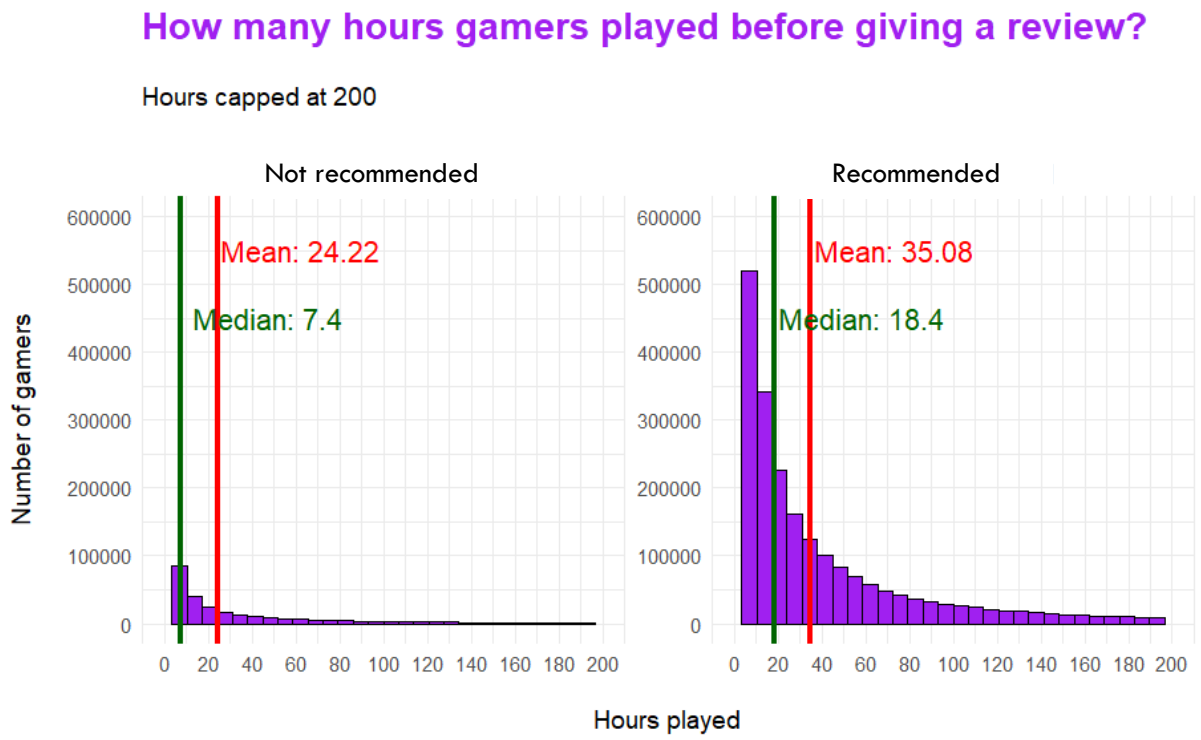
DATA EXPLORATION

First step:

- Trimming the dataset → keeping the observations from June 2022 to December 2022

Are reviews reliable? After how many hours did players leave a review?

Negative reviews are written much sooner than positive ones.

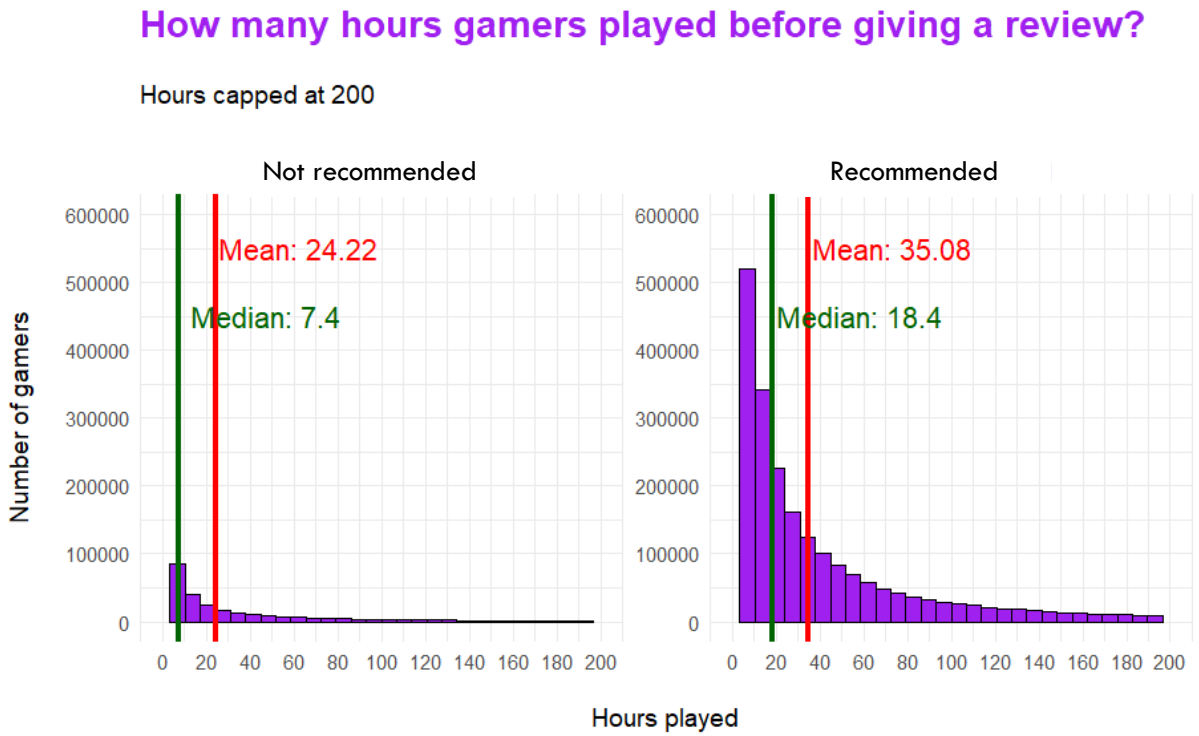


PREPROCESSING – DEALING WITH NEGATIVE REVIEWS

If a gamer leaves a negative review:

a. They played the game for less than the third quartile[1] → the review is registered as negative.
b. They played the game for more than the third quartile → the review is considered positive.

Key point:
You will not play a game you do not like.



[1]third quartile: computed over the all number of hours, corresponding to 71 hours.

PREPROCESSING – CREATING THE GRAPH

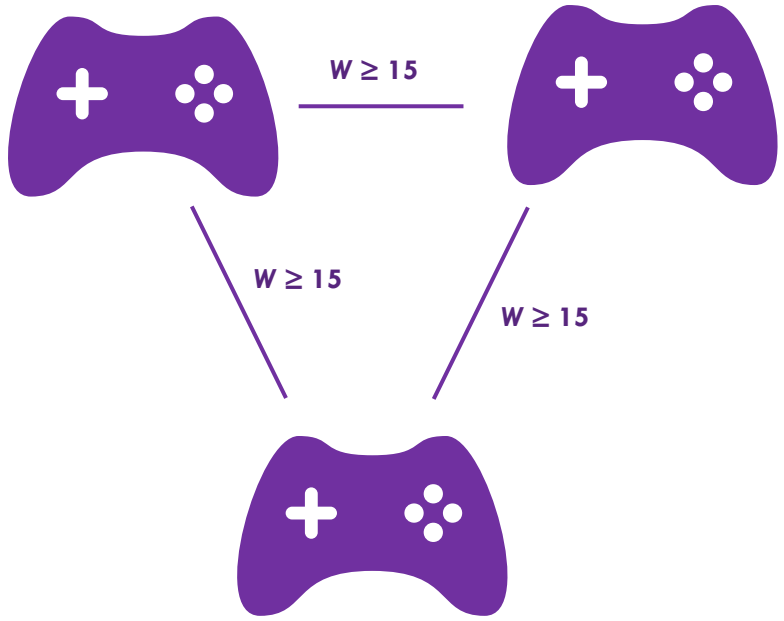
Then:

- Filter out *non-popular* games
- Filter out *non-recommended* games

Graph structure:

- Nodes: games
- Edges: two nodes will be linked if they have been positively reviewed by at least 15 players
- Weight: how many times two games have been recommended, at least 15

The graph is not oriented.



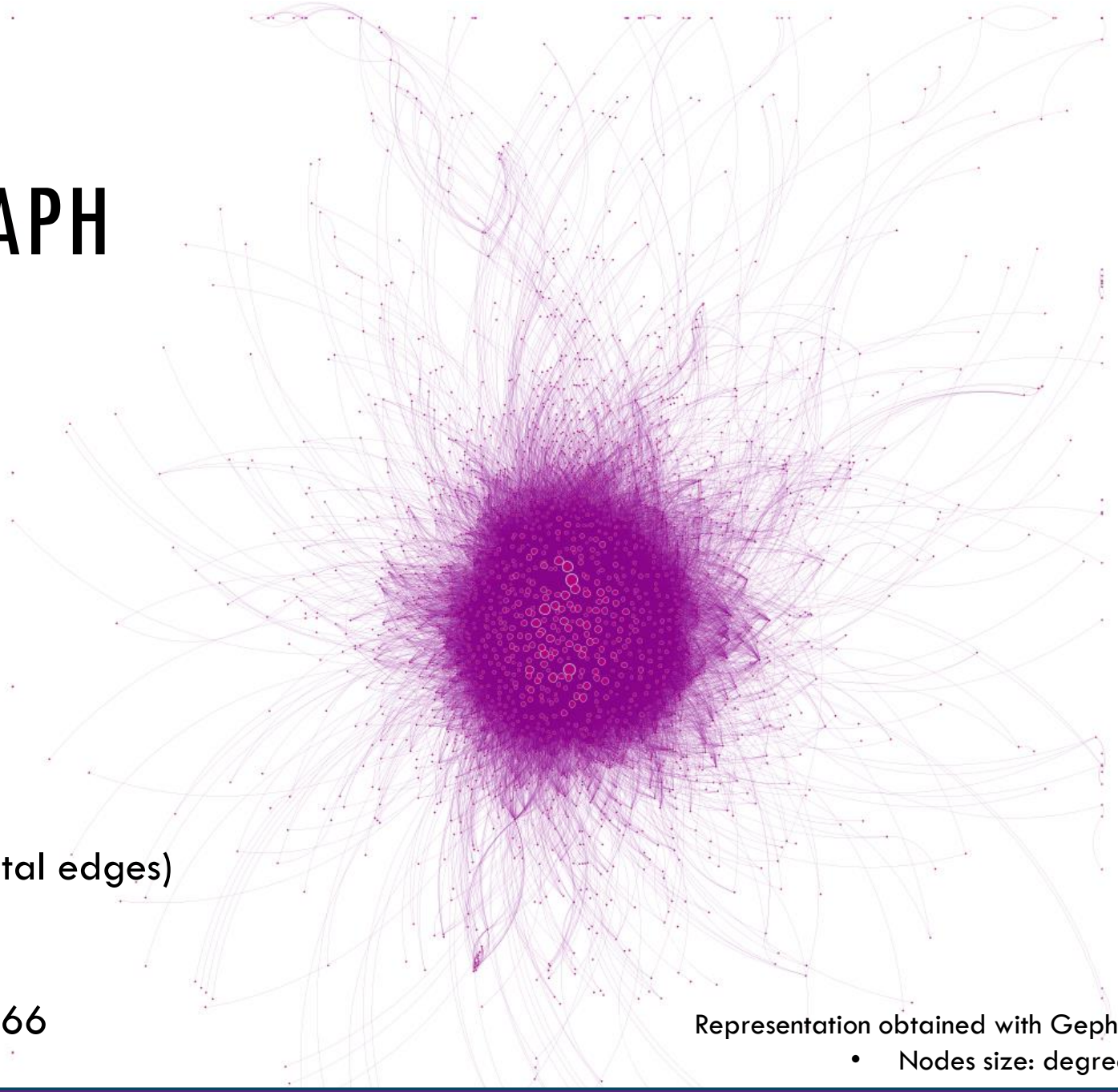
ANALYSING THE GRAPH

Basic characteristics:

- 2356 nodes
- 60,115 edges

Graph properties:

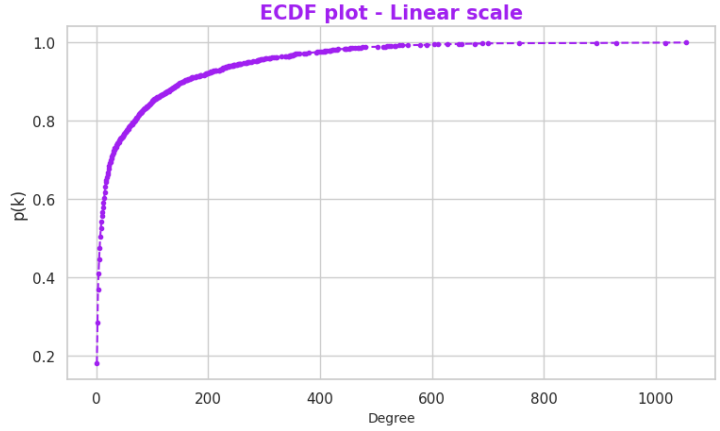
- Max degree: 1054
- Min degree: 1
- Average degree: 51.03
- Standard deviation: 106.50
- Number of connected components: 68
- Number of bridges: 387 (0.64% of total edges)
- Network density: 0.022
- Global clustering coefficient: 0.38
- Average local clustering coefficient: 0.66

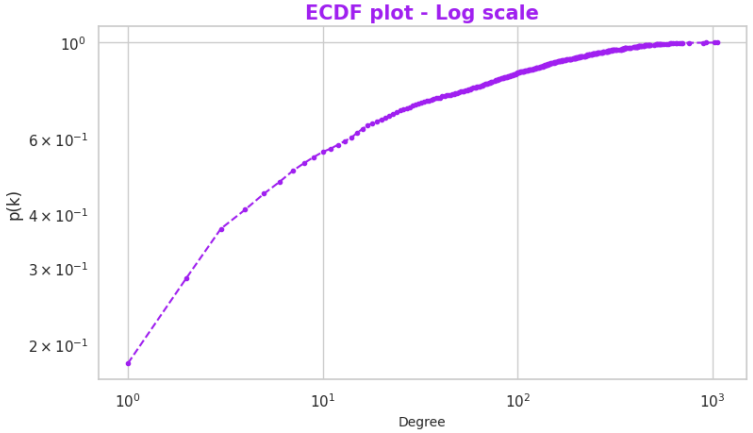


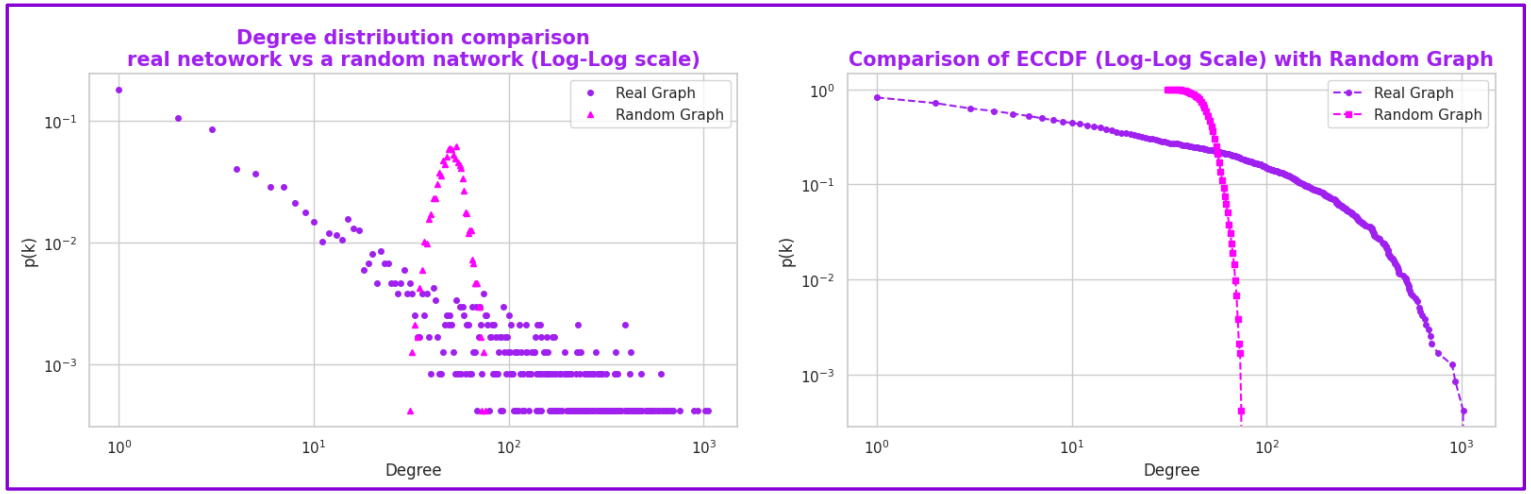
Representation obtained with Gephi:

- Nodes size: degree

ANALYSING THE GRAPH – ECDF AND ECCDF







Unsurprisingly we have scale free behaviour!

ANALYSING THE GRAPH – CENTRALITIES

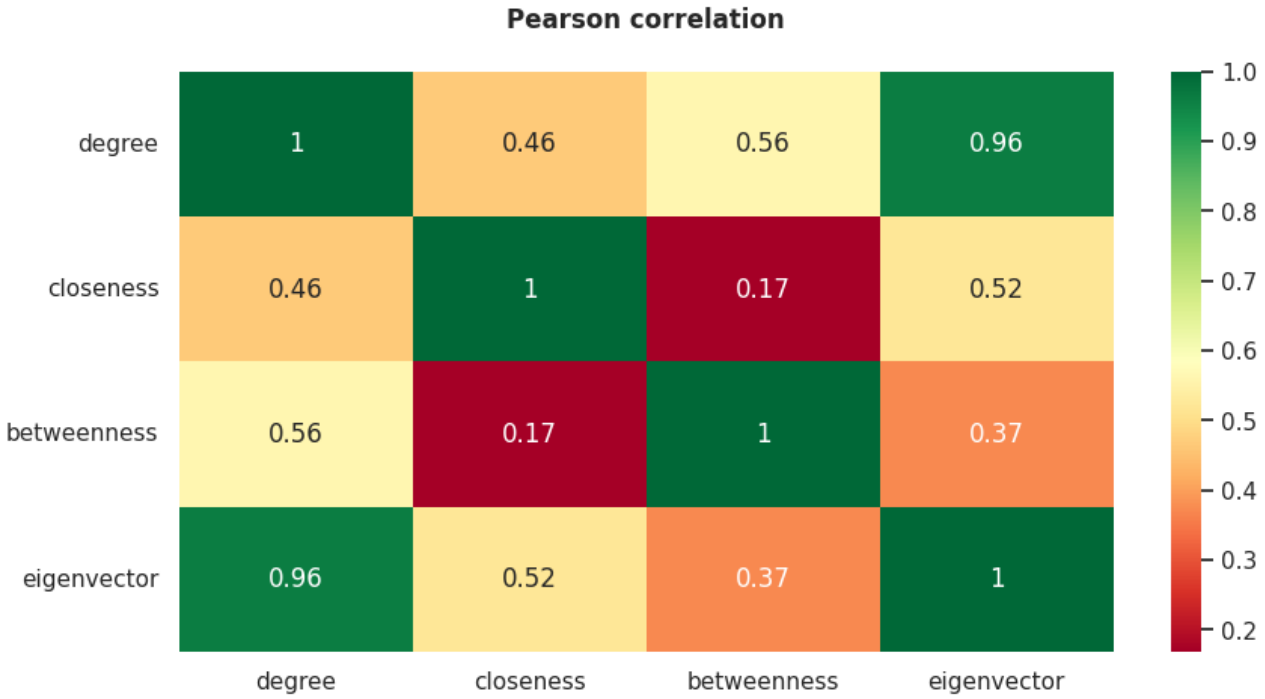
Top three games for each centrality measure:

Degree:
1. Stray
2. Left 4 Dead 2
3. Vampire Survivors

Closeness:
1. Stray
2. Left 4 Dead 2
3. Vampire Survivors

Betweeness:
1. Stray
2. Left 4 Dead 2
3. Vampire Survivors

Eigenvector:
1. Left 4 Dead 2
2. Stray
3. Call of Duty

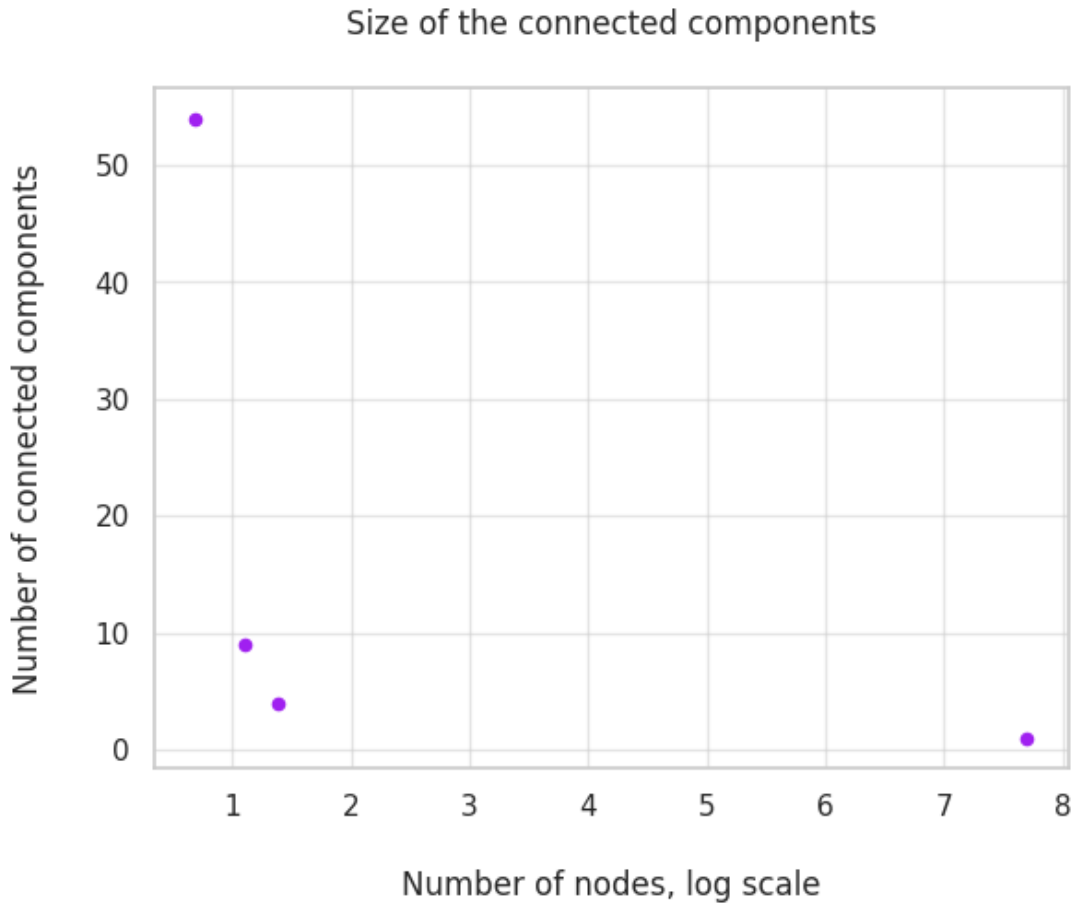


ANALYSING THE GRAPH – CONNECTED COMPONENTS

The graph is not connected and contains 68 connected components.

The largest connected component:

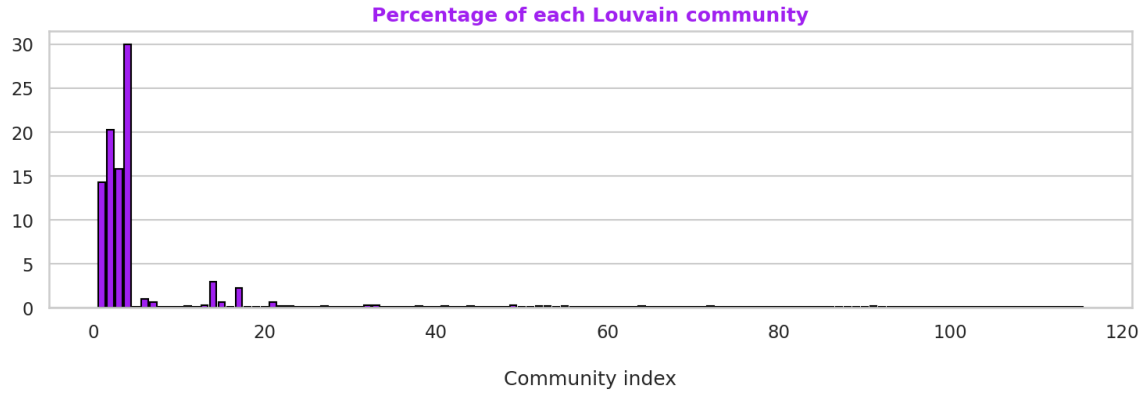
- Number of nodes 2205 nodes (93.5% of total)
- Number of edges: 60,014 (99.83% of total)
- Average degree: 27.22
- Average shortest path: 2.58

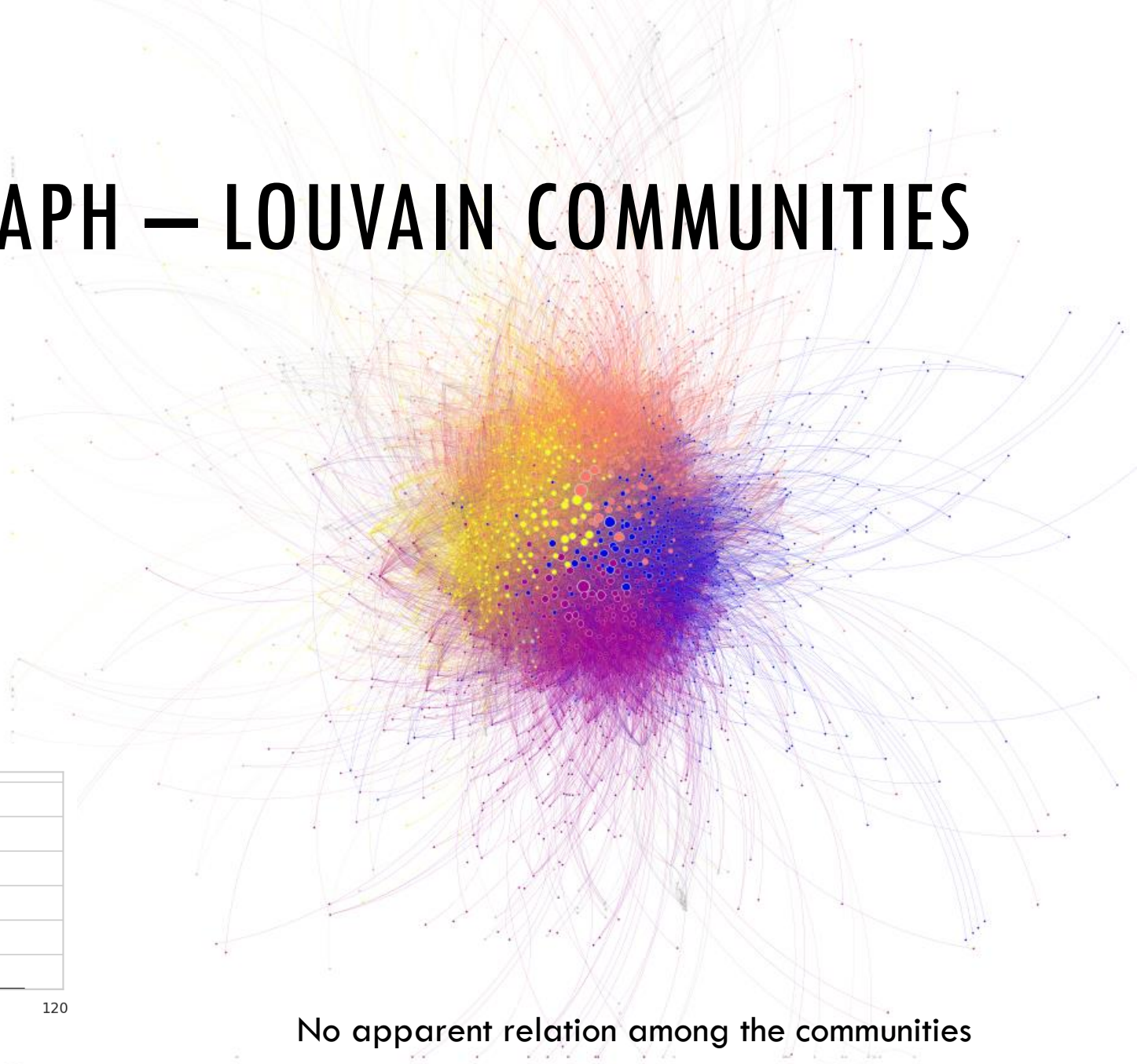


ANALYSING THE GRAPH – LOUVAIN COMMUNITIES

A total of 105 communities found and the most frequent are:

- 4 → 30%
- 2 → 20%
- 3 → 15.79%
- 1 → 14.26%
- All the others → 19.95%





No apparent relation among the communities

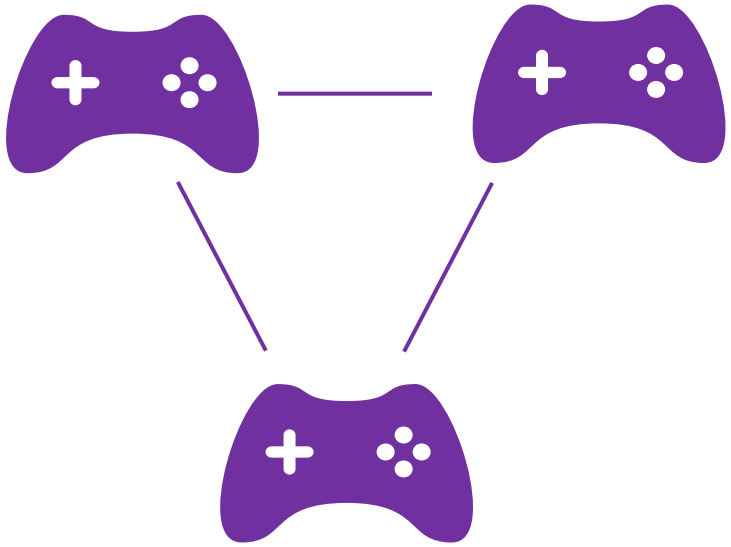
NODE2VEC THEORY

Node2Vec is an unsupervised learning algorithm working with the main ideas of DeepWalk with a twist:

- Biased random walks → parameters **p** and **q** to tune in order to obtain better results
- Word2Vec for node embeddings

Two main samping strategies:

- **Breadth-First Search (BFS)** → considering the closest nodes in terms of connections, the "neighbourhood" (controlled by the parameter **q**)
- **Depth-First Search (DFS)** → considering nodes that are not adjacent to the previous nodes (controlled by the parameter **p**). This allows to have a more macro view of the graph.



NODE2VEC IMPLEMENTATION

- Same game, same position
- Same game, different position

DFS → p=4, q=1		BFS → p=1, q=4	
ArmA 3:	**Portal 2:**	**ArmA 3:**	**Portal 2:**
1. ArmA 2	1. Portal	1. ArmA 2	1. Portal
2. Squad	2. Portal Stories: Mel	2. Squad	2. Half-life
3. DCS World Steam Edition	3. Half-Life 2	3. Insurgency: Sandstorm	3. Half-Life 2
For the King:	**Outlast:**	**For the King:**	**Outlast:**
1. Gunfire Reborn	1. Outlast 2	1. Gunfire Reborn	1. Outlast 2
2. Core Keeper	2. Slander: the Arrival	2. PlateUp!	2. Dying light
3. PlateUp!	3. Dying light	3. Tabletop simulator	3. Little nightmares

CONCLUSIONS

Working with graphs is extremely expensive, indeed to obtain something interesting I kept a miniscule fraction of what I had at the beginning, around 41 million reviews.

The Node2Vec algorithm performs well, predicting games of the same genre and/or of the same saga.

However, Steam recommandation algorithm is much more complex, using also information of game genres, hours played – not the ones before giving a review – and much more. Still, in some cases Node2Vec was able to suggest the same games the Steam algorithm suggests.

Thank you for the attention!