Investment Analytics Using Clustering Algorithms

Ali Fardaev
Isaac Gonzalez
Safiya Alavi
Samuel Dominguez

DS 205

AGENDA

01 **BUSINESS CASE SCENARIO**

INVESTMENT PORTFOLIO ANALYTICS

02 **NEO4J USE & LIVE DEMO**

LOUVAIN MODULARITY
LEIDEN ALGORITHM
PAGERANK

03 **MONGODB USE**

USE CASE DISCUSSION

04 **REDIS USE**

USE CASE DISCUSSION

05 **CONCLUSION**

QUESTIONS
THANK YOU

QUANTARI
Smarter Investment Decisions

Business Opportunity:

- Investors hold high-performing stocks but risk becoming over-concentrated.

- Diversify to mitigate risk.
  - Identify stocks with similar 90-day performance

- Strategic advantage ⇒ Investment in potentially undervalued companies sharing key market characteristics with proven winners in the portfolio.

QUANTARI
Smarter Investment Decisions

Proposed Product:

- User-friendly interface

- Clusters stocks based on similarity in performance
  - Similarity measured with:
    - Pearson's correlation coefficient (to dollar changes in daily share price)
      - implemented in platform
    - Cosine similarity
      - tested but not implemented
    - Minimum Spanning Tree
      - tested but not implemented

- Clustering methods both identifies stocks with similar correlation values and provides diversification options.
  - Neo4J graphs and clustering algorithms used to achieve this.

NEO4J GRAPH DATABASE AND GRAPHS ALGORITHMS

NEO4J GRAPH DATABASE

- Graph database management system
- Stores and analyzes highly connected data through nodes and relationships
- Enables querying and visualization of these data structures.

LOUVAIN MODULARITY

- Hierarchical clustering algorithm
- Directed and Undirected graph
- Maximizes a modularity score for each community
  - Measures density of connections within community versus overall random network

LEIDEN ALGORITHM

- Hierarchical clustering algorithm
- Undirected graphs
- Addresses some shortcomings in Louvain:
  - Algorithm randomly breaks down communities into smaller well-connected ones.

PAGERANK

- Measures the importance of each node within the graph
- Assumption that a page is only as important as the pages that link to it
- Directed and Undirected graphs

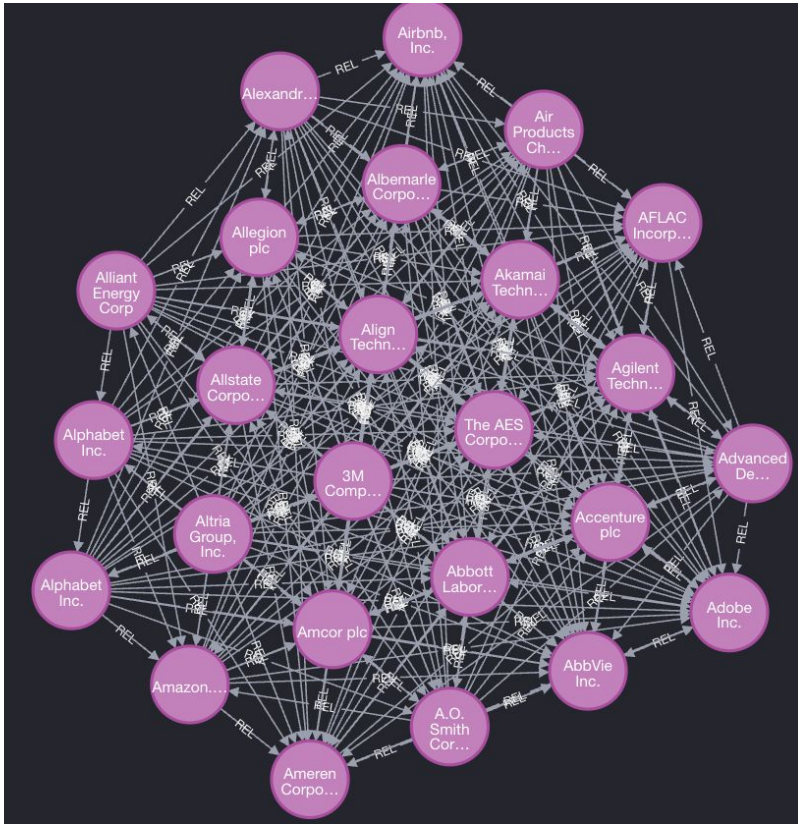
QUANTARI

Smarter Investment Decisions

Link to site:
http://127.0.0.1:5000/

GRAPH DESIGN

- Nodes represent individual stock tickers.
- Relationships weighted using Pearson correlation coefficients.
  - Edges with negative weights are dropped
  - Positive weights multiplied by 1,000
- Graph scale is fully connected
  - 500 nodes
  - >126,000 undirected relationships.
  - Structure is optimized for identifying clusters of related stocks.



Screenshot of our graph in Neo4j browser showing 25 stocks and their relationships

USE OF LEIDEN/LOUVAIN ALGORITHM FOR BUSINESS SCENARIO

- Louvain and Leiden are used for community detection.
- This offers a unique diversification or similarity strategy
  - The former preventing overconcentration risks
  - The latter helping to focus returns.
- This works on the graph's weighted, undirected edges.
  - Leiden Algorithm is basically a better version of Louvain because it's less sensitive to creating fewer, larger clusters.
  - Both algorithms are useful for identifying groups of related stocks
    - Given a query stock, we're able to show stocks similar to the query and other groups that are dissimilar.

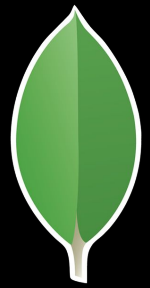
USE OF PAGERANK FOR BUSINESS SCENARIO

$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)},$$

https://en.wikipedia.org/wiki/PageRank

- Platform returns top n stocks of the PageRank ranking to the target stock.
  - Planned for the next iteration: an alternative approach to a diversification strategy and minimizes overconcentration risks by
    - Showing N stocks near the 50th percentile
    - Showing N stocks near the 1st percentile
- The algorithm ranks a query stock and the connected edges by ‘quality’, which is a probabilistic measure, using the edge weights in the graph.

- Relational databases do not directly support graph algorithms
- Linking nodes with relationships are easier than managing multiple tables and tracking them .
- Use of built-in algorithms in Neo4j are more efficient
- A very large request may take up lots of memory on a relational database.



mongoDB

User portfolio history

- Schemaless structure = Flexible user profiles
- Easily add new fields

- Better than relational database because no need to have different tables, joints, or migrations

redis

Caching graph queries

- Avoid costly recomputation of graph algos
- Real time suggestions

- Better than relational database because:
  - Disk is much slower than in-memory Redis
  - Native Key-Value caching

THANK YOU

Questions?