Task 1: Network Construction & Preliminaries

1. Dataset Description:

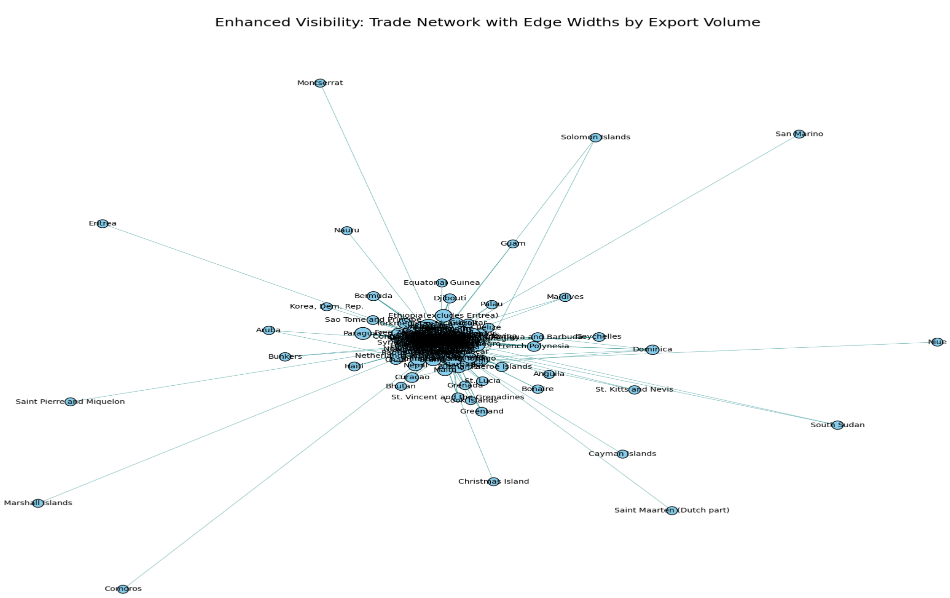
- **Nodes**: Countries.
- **Edges**: Export relationships . An edge from Country A to B means A exports to B.
- **Edge Weight**: Volume of exports of minerals in USD (thousands).
- **Source**: [Link]

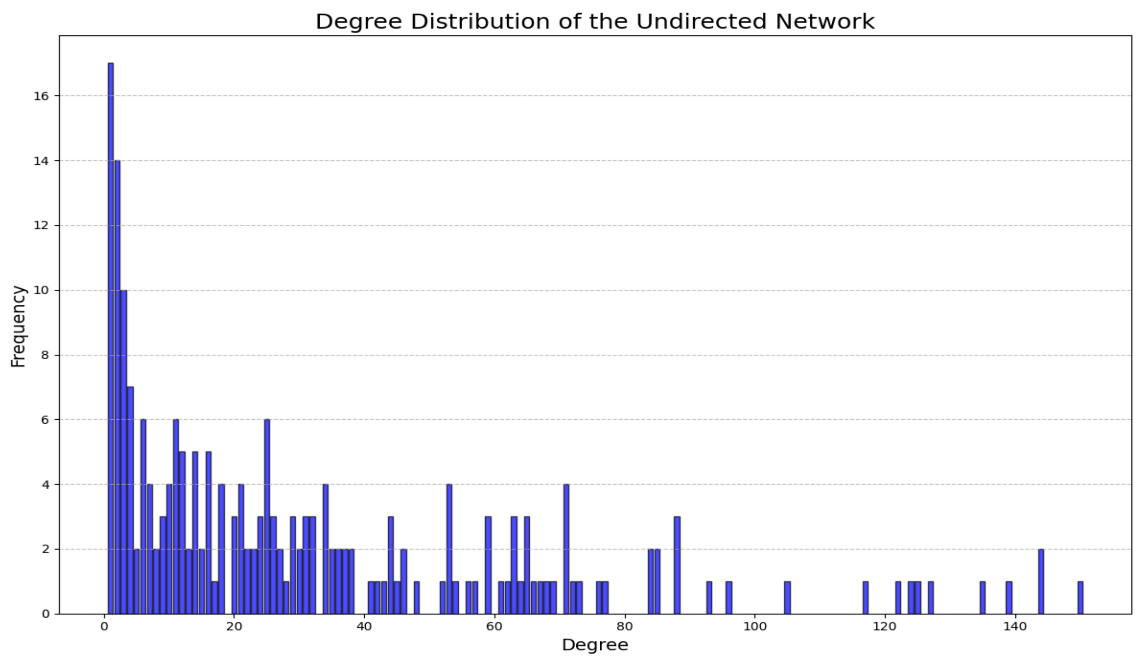
2. Construct Your Network:

- Created an undirected**, weighted network** using NetworkX.
- Edges were added only if export volume > 0.

3. Initial Observations:

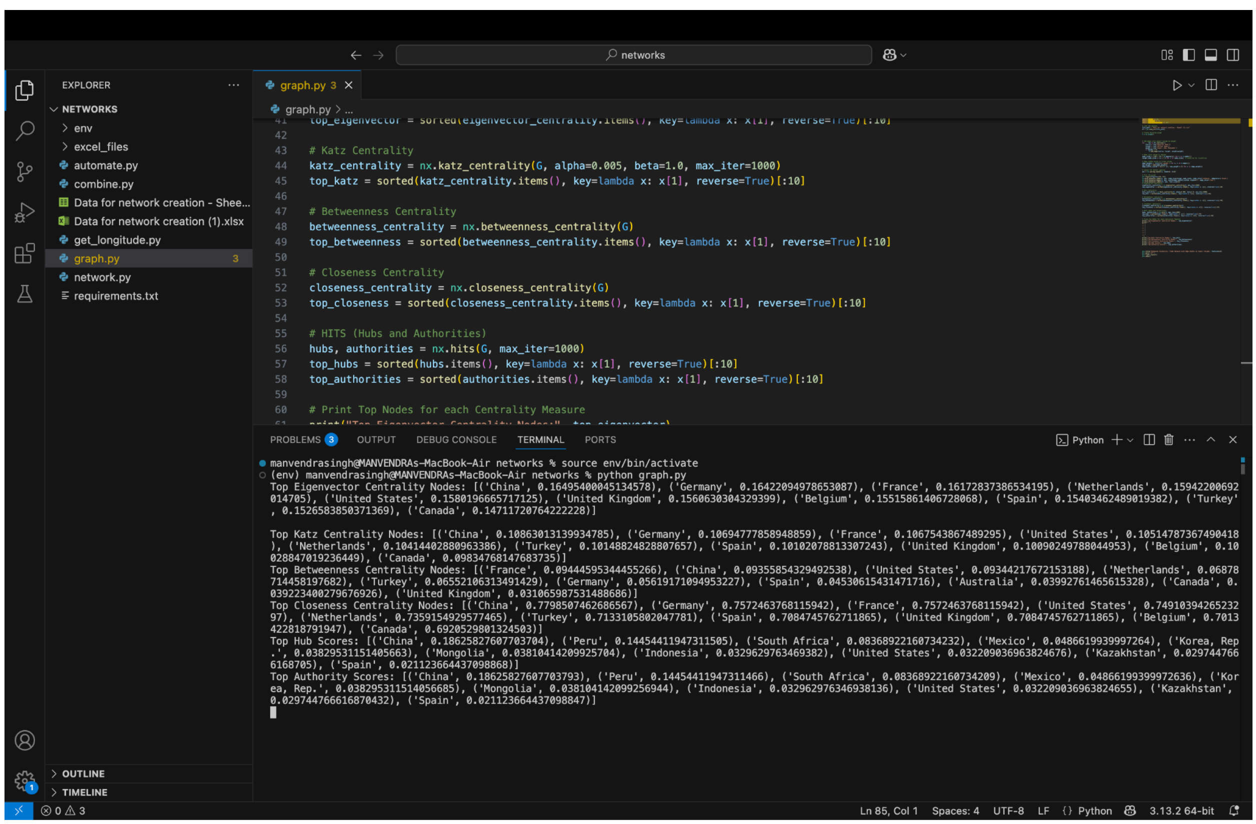
- **Number of Nodes (Countries):** 222
- **Number of Edges (Export Relationships):** 5916





Degree Distribution of the Undirected Network
Frequency
Degree

Task 2



networks
graph.py 3
EXPLORER
NETWORKS
env
excel_files
automate.py
combine.py
Data for network creation - Shee...
Data for network creation (1).xlsx
get_longitude.py
graph.py
network.py
requirements.txt
OUTLINE
TIMELINE
# Katz Centrality
katz_centrality = nx.katz_centrality(G, alpha=0.005, beta=1.0, max_iter=1000)
top_katz = sorted(katz_centrality.items(), key=lambda x: x[1], reverse=True)[:10]
# Betweenness Centrality
betweenness_centrality = nx.betweenness_centrality(G)
top_betweenness = sorted(betweenness_centrality.items(), key=lambda x: x[1], reverse=True)[:10]
# Closeness Centrality
closeness_centrality = nx.closeness_centrality(G)
top_closeness = sorted(closeness_centrality.items(), key=lambda x: x[1], reverse=True)[:10]
# HITS (Hubs and Authorities)
hubs, authorities = nx.hits(G, max_iter=1000)
top_hubs = sorted(hubs.items(), key=lambda x: x[1], reverse=True)[:10]
top_authorities = sorted(authorities.items(), key=lambda x: x[1], reverse=True)[:10]
# Print Top Nodes for each Centrality Measure
PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS
manvendrasingh@MANVENDRAs-MacBook-Air networks % source env/bin/activate
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks % python graph.py
Top Eigenvector Centrality Nodes: [('China', 0.16495400045134578), ('Germany', 0.16422094978653087), ('France', 0.16172837386534195), ('Netherlands', 0.15942200692014705), ('United States', 0.1580196665717125), ('United Kingdom', 0.1560630304329399), ('Belgium', 0.15515861406728068), ('Spain', 0.15403462489019382), ('Turkey', 0.1526583850371369), ('Canada', 0.14711720764222228)]
Top Katz Centrality Nodes: [('China', 0.10863013139934785), ('Germany', 0.10694777858948859), ('France', 0.1067543867489295), ('United States', 0.10514787367490418), ('Netherlands', 0.10414402880963386), ('Turkey', 0.10148824828807657), ('Spain', 0.10102078813307243), ('United Kingdom', 0.10090249788044953), ('Belgium', 0.10028847019236449), ('Canada', 0.09834768147683735)]
Top Betweenness Centrality Nodes: [('France', 0.09444595344455266), ('China', 0.09355854329492538), ('United States', 0.09344217672153188), ('Netherlands', 0.06878714458197682), ('Turkey', 0.06552106313491429), ('Germany', 0.05619171094953227), ('Spain', 0.04530615431471716), ('Australia', 0.03992761465615328), ('Canada', 0.039223400279676926), ('United Kingdom', 0.031065987531488686)]
Top Closeness Centrality Nodes: [('China', 0.7798507462686567), ('Germany', 0.7572463768115942), ('France', 0.7572463768115942), ('United States', 0.7491039426523297), ('Netherlands', 0.7359154929577465), ('Turkey', 0.7133105802047781), ('Spain', 0.7084745762711865), ('United Kingdom', 0.7084745762711865), ('Belgium', 0.7013422818791947), ('Canada', 0.6920529801324503)]
Top Hub Scores: [('China', 0.18625827607703704), ('Peru', 0.14454411947311505), ('South Africa', 0.08368922160734232), ('Mexico', 0.0486619939997264), ('Korea, Rep.', 0.03829531151405663), ('Mongolia', 0.03810414209925704), ('Indonesia', 0.0329629763469382), ('United States', 0.032209036963824676), ('Kazakhstan', 0.0297447666168705), ('Spain', 0.021123664437098868)]
Top Authority Scores: [('China', 0.18625827607703793), ('Peru', 0.14454411947311466), ('South Africa', 0.08368922160734209), ('Mexico', 0.04866199399972636), ('Korea, Rep.', 0.038295311514056685), ('Mongolia', 0.038104142099256944), ('Indonesia', 0.032962976346938136), ('United States', 0.032209036963824655), ('Kazakhstan', 0.029744766616870432), ('Spain', 0.021123664437098847)]
Ln 85, Col 1 Spaces: 4 UTF-8 LF Python 3.13.2 64-bit

•

1. Eigenvector Centrality

This measures influence—nodes connected to highly connected nodes score higher.

Top 10 Nodes:

1. China
2. Germany
3. France
4. Netherlands
5. United States
6. United Kingdom
7. Belgium
8. Spain
9. Turkey
10. Canada

These are core economies with extensive trade connections to other central and influential economies.

🔹 2. Katz Centrality

Like eigenvector centrality but also gives credit for being connected to less influential nodes.

Top 10 Nodes:

1. China
2. Germany
3. France
4. United States
5. Netherland
6. Turkey
7. Spain
8. United Kingdom
9. Belgium
10. Canada

These are not only connected to other central players but also to peripheral ones, increasing their overall influence.

🔹 3. Betweenness Centrality

Measures how often a node lies on the shortest path between other nodes—key for trade routing.

Top 10 Nodes:

1. France
2. China
3. United States
4. Netherland
5. Turkey
6. Germany
7. Spain
8. Australia
9. Canada
10. United Kingdom

🔹 4. Closeness Centrality

Reflects how quickly a node can interact with others—short average path lengths.

Top 10 Nodes:

1. China
2. Germany
3. France
4. United States
5. Netherlands
6. Turkey
7. Spain
8. United Kingdom
9. Belgium

10. Canada

🔹 5. HITS Algorithm

Separates roles: **Hubs** (exporters to authorities) and **Authorities** (receivers of attention from hubs).

▪ **Hubs (Top Export Nodes)**

1. China
2. Peru
3. South Africa
4. Mexico
5. Korea Rep.
6. Mongolia
7. Indonesia
8. United States
9. Kazakhstan
10. Spain

▪ **Authorities (Top Import Nodes)**

1. China
2. Peru
3. South Africa
4. Mexico
5. Korea, Rep.
6. Mongolia
7. Indonesia
8. United States
9. Kazakhstan
10. Spain

Comparison Across Centralities

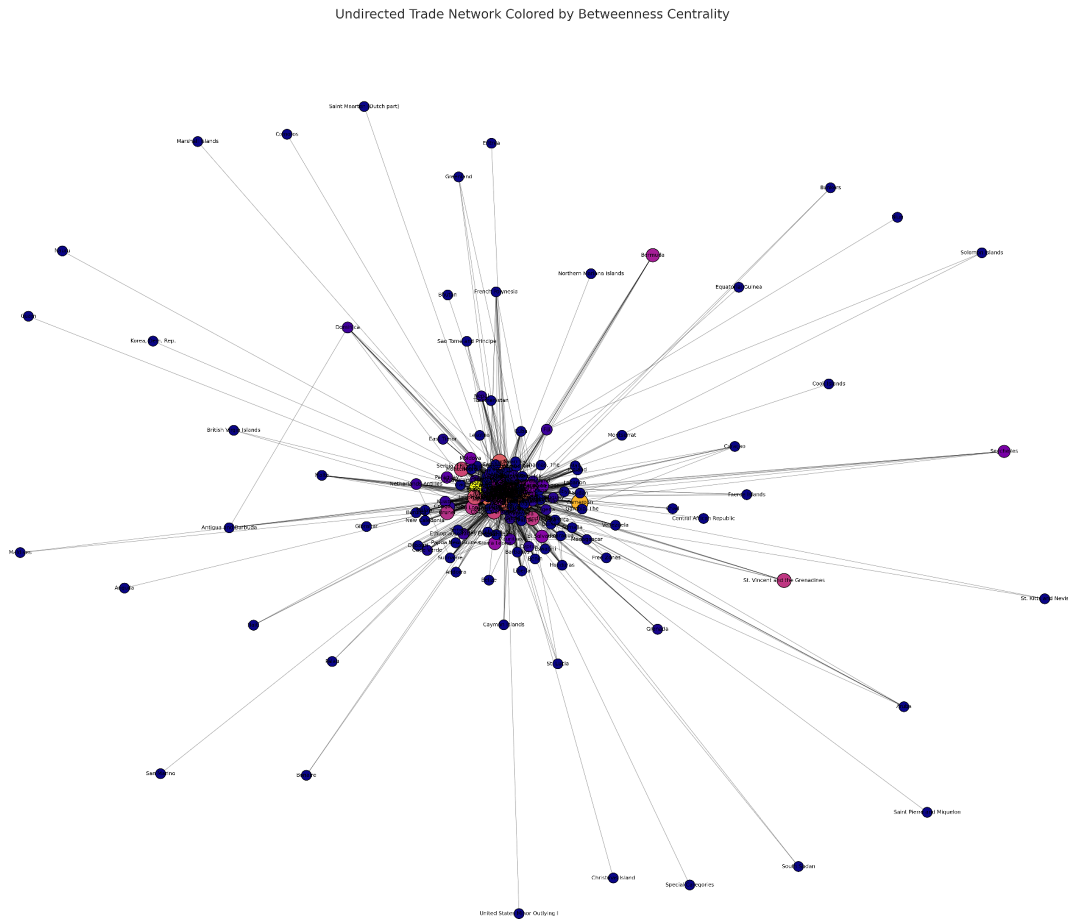
Country	Eigenvector	Katz	Betweenness	Closeness	HITS-Hub	HITS-Authority
China	✅	✅	✅	✅	✅	✅
Germany	✅	✅	❌	✅	❌	❌
France	✅	✅	✅	✅	❌	❌
USA	✅	✅	✅	✅	✅	✅
Netherlands	✅	✅	✅	✅	❌	❌

Consistent Leaders: Countries like China, Germany, France, and the United States rank high across multiple metrics, confirming their central roles in global trade.

Metric-Specific Insights: Betweenness centrality and HITS differentiate countries based on trade routing and role specialization (exporter vs. importer).

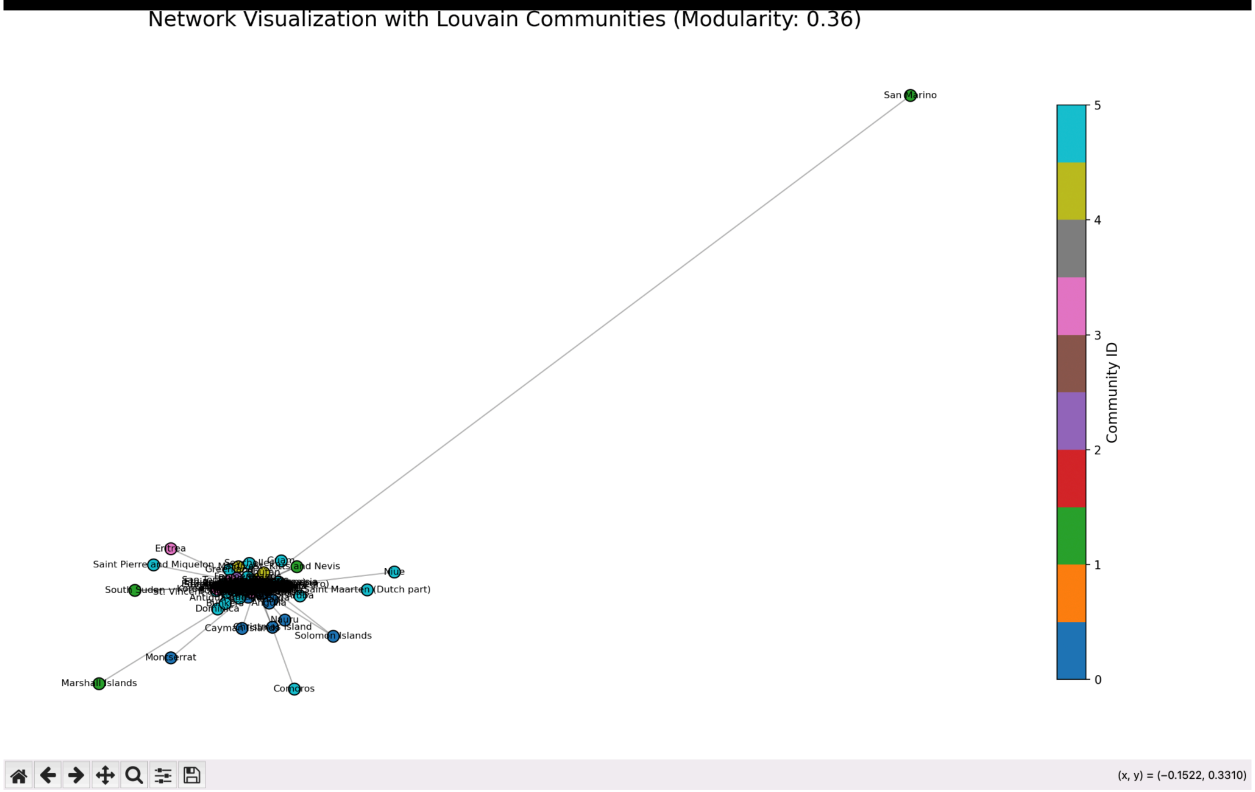
Diverse Roles: While some countries act as both exporters and importers (e.g., China), others like Turkey focus on bridging connections within the network.

Policy Implications: Understanding these centralities can guide trade negotiations, regional partnerships, and investments in key economies.



Here's the **visualization of your undirected trade network**, with nodes colored and sized by **betweenness centrality**. You can clearly see which countries are structurally important in terms of connecting trade flows.

Task 3




```
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks % python graph.py
Number of communities detected: 6
Modularity score: 0.36

Community Partition:
Community 1: ['Brazil', 'United States', 'Argentina', 'Australia', 'Bolivia', 'Canada', 'C
hile', 'Japan', 'Korea, Rep.', 'Mexico', 'New Zealand', 'Paraguay', 'Uruguay', 'Curaçao',
'Christmas Island', 'Nauru', 'Solomon Islands', 'Bahrain', 'Algeria', 'Bahamas, The', 'Jam
aica', 'Costa Rica', 'Madagascar', 'Bermuda', 'Cuba', 'Haiti', 'New Caledonia', 'Cayman Is
lands', 'Mauritius', 'Cook Islands', 'Anguila', 'Montserrat', 'Netherlands Antilles']
Community 2: ['Bulgaria', 'Bosnia and Herzegovina', 'Greece', 'Croatia', 'North Macedonia'
, 'Montenegro', 'Poland', 'Serbia, FR(Serbia/Montenegro)', 'Turkey', 'Armenia', 'Georgia',
 'Ukraine', 'Ghana', 'Suriname', 'Cameroon', 'Cyprus', 'Czech Republic', 'Hungary', 'Israe
l', 'Nigeria', 'Romania', 'Slovak Republic', 'Barbados', 'Guyana', 'St. Lucia', 'Moldova',
 "Cote d'Ivoire", 'Dominican Republic', 'Sierra Leone', 'Syrian Arab Republic', 'Burkina F
aso', 'Mali', 'Chad', 'Liberia', 'St. Kitts and Nevis', 'Occ.Pal.Terr', 'South Sudan', 'Ma
rshall Islands', 'San Marino']
Community 3: ['Ethiopia(excludes Eritrea)', 'Djibouti']
Community 4: ['China', 'Namibia', 'Russian Federation', 'Vietnam', 'Indonesia', 'Peru', 'B
elarus', 'Iran, Islamic Rep.', 'Ecuador', 'Gabon', 'Lao PDR', 'Mongolia', 'Malaysia', 'Oth
er Asia, nes', 'Philippines', 'Papua New Guinea', 'East Timor', 'Azerbaijan', 'Panama', 'K
azakhstan', 'Uzbekistan', 'Iraq', 'Kenya', 'Mauritania', 'Brunei', 'Nicaragua', 'Eritrea',
 'Kyrgyz Republic', 'Macao', 'Korea, Dem. Rep.', 'Tajikistan', 'Turkmenistan', 'Palau']
Community 5: ['India', 'Egypt, Arab Rep.', 'Saudi Arabia', 'Thailand', 'United Arab Emirat
es', 'Bangladesh', 'Guinea', 'Jordan', 'Oman', 'Pakistan', 'Guatemala', 'Lebanon', 'Sri La
nka', 'Morocco', 'Nepal', 'Rwanda', 'Kuwait', 'Colombia', 'Honduras', 'Togo', 'Belize', 'B
enin', 'El Salvador', 'Venezuela', 'Burundi', 'Cambodia', 'Afghanistan', 'Myanmar', 'Somal
ia', 'Sudan', 'Free Zones', 'Maldives', 'South Asia', 'Bhutan', 'Yemen']
Community 6: ['Albania', 'Austria', 'Belgium', 'Germany', 'Spain', 'France', 'Italy', 'Net
herlands', 'Andorra', 'Angola', 'Congo, Rep.', 'United Kingdom', 'Portugal', 'Sao Tome and
 Principe', 'Denmark', 'Finland', 'Hong Kong, China', 'Switzerland', 'Fiji', 'Norway', 'Qa
tar', 'Senegal', 'Singapore', 'Slovenia', 'Sweden', 'Tanzania', 'Estonia', 'Ireland', 'Ice
land', 'Lithuania', 'Luxembourg', 'Tunisia', 'Uganda', 'Latvia', 'Bonaire', 'Botswana', 'M
alta', 'Mozambique', 'Trinidad and Tobago', 'Niger', 'Greenland', 'Cape Verde', 'Gambia, T
he', 'Libya', 'Malawi', 'Congo, Dem. Rep.', 'Faeroe Islands', 'Dominica', 'Antigua and Bar
buda', 'Bunkers', 'Eswatini', 'Niue', 'Lesotho', 'Sub-Saharan Africa', 'French Polynesia',
 'St. Vincent and the Grenadines', 'Comoros', 'Saint Pierre and Miquelon', 'Saint Maarten
(Dutch part)', 'Seychelles', 'Grenada', 'Gibraltar', 'Equatorial Guinea', 'Zimbabwe', 'Aru
ba', 'Guam', 'South Africa', 'Zambia']
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks %
Ln 36, Col 13   Spaces: 4   UTF-8   LF   {} Python   3.13.2 64-bit
```

Here Here are the results from the **Girvan–Newman community detection**:

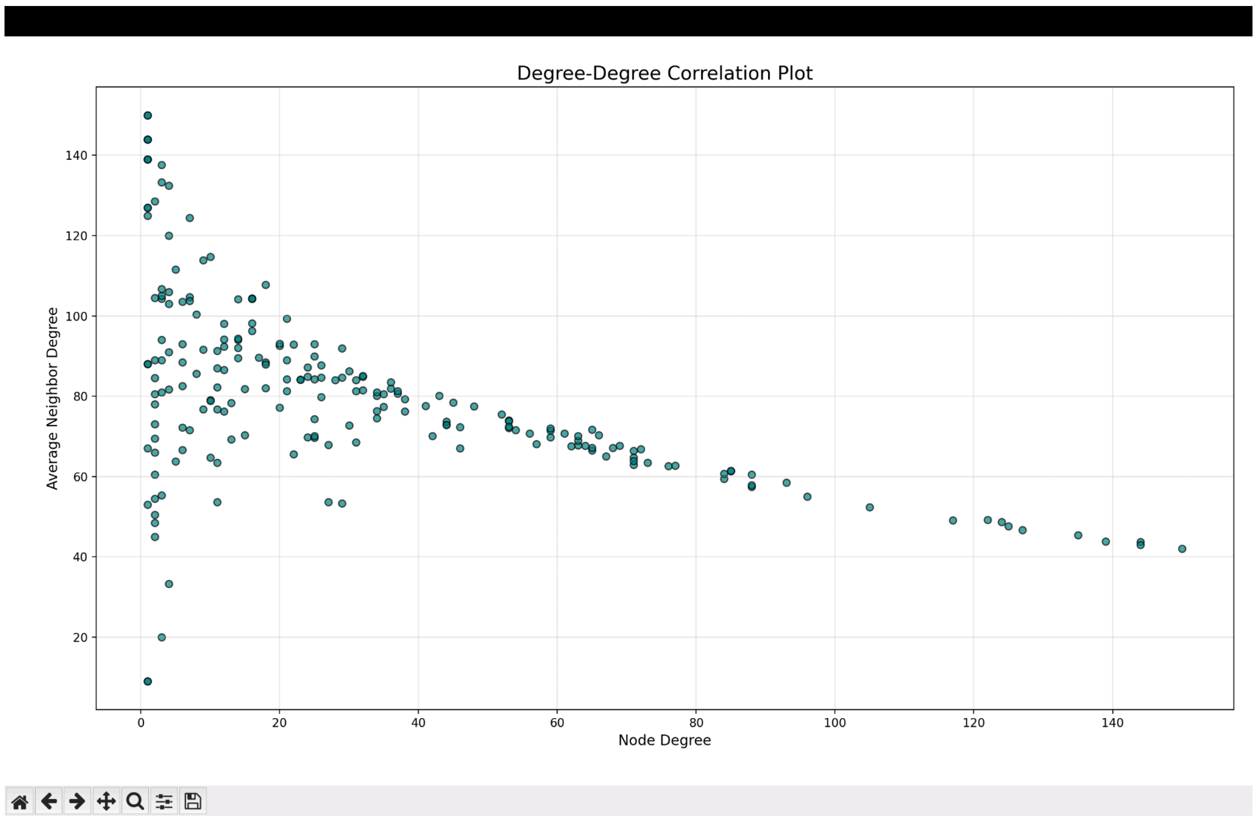
- **Number of Communities:** 6

- **Modularity Score:** ~0.36

A modularity score of **0.36** indicates that the network has a well-defined community structure

Task 4

Results:

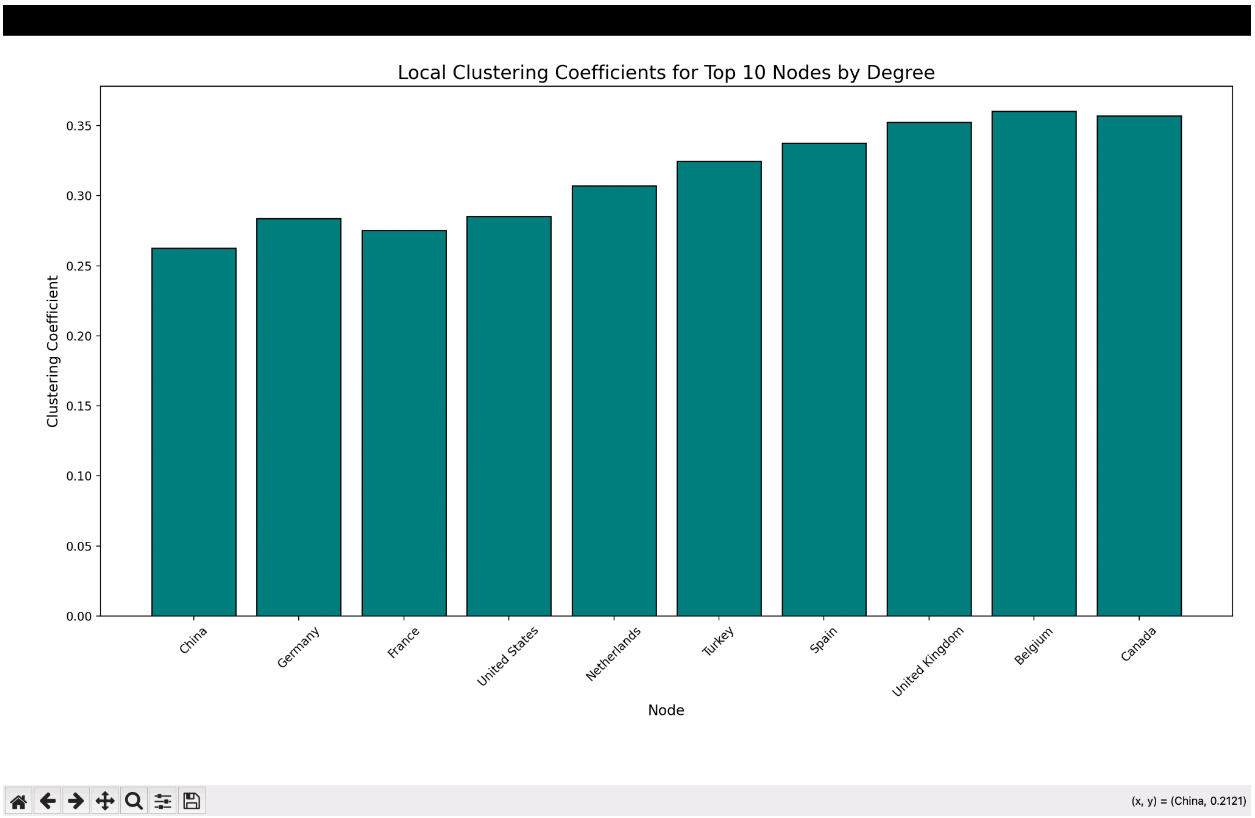


- **Degree Assortativity Coefficient (Pearson Correlation): -0.34**

Interpretation:

- Since the value is **negative**, your network is **disassortative**.
- This means that **high-degree nodes tend to connect to low-degree nodes** rather than to other high-degree nodes.
- Such patterns are common in **trade networks**, where hubs (large exporters/importers) interact with many smaller partners.

Task 5



🔍 Clustering Coefficient Results:

```
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks % python graph.py
Degree Assortativity (Pearson Correlation Coefficient): -0.34
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks % python graph.py
Degree Assortativity (Pearson Correlation Coefficient): -0.34
(env) manvendrasingh@MANVENDRAs-MacBook-Air networks % python graph.py
Global Clustering Coefficient (Transitivity): 0.5069
Random Graph Clustering Coefficient: 0.1572

Local Clustering Coefficients for Top 10 Nodes by Degree:
China: 0.2626
Germany: 0.2839
France: 0.2753
United States: 0.2854
Netherlands: 0.3071
Turkey: 0.3246
Spain: 0.3375
United Kingdom: 0.3525
Belgium: 0.3602
Canada: 0.3571
```

⊗ 0 ⚠ 3   Ln 54, Col 1   Spaces: 4   UTF-8   LF   {} Python   3.13.2 64-bit

- Global Clustering Coefficient (Trade Network): 0.5069
- Global Clustering Coefficient (Random Graph): 0.1572

The global clustering coefficient is significantly higher than the random graph's, indicating a strong clustering tendency in the network.