

```

import networkx as nx
import urllib.request as u
from google.colab import files

def create_networks():
    """Create both directed and undirected networks from URL data"""
    url = "https://drive.google.com/file/d/1k9shqEUbTg1tL3-Y0vtLrt1VF3Bqnr9-/view?usp=sharing"
    file_id = url.split('/')[-2]
    download_url = f"https://drive.google.com/uc?id={file_id}"

    # Create empty graphs
    G = nx.Graph()          # undirected
    D = nx.DiGraph()        # directed

    # Read and add edges
    response = u.urlopen(download_url)
    data = response.read().decode('utf-8')

    for line in data.splitlines():
        if line and not line.startswith('#') and not
line.startswith('%'):
            try:
                s, t, w = map(float, line.strip().split())
                G.add_edge(int(s), int(t), weight=w)
                D.add_edge(int(s), int(t))
            except:
                continue

    return G, D

def export_for_cytoscape(G, D):
    """Export networks as CSV for Cytoscape visualization"""
    # Export undirected weighted network
    with open('network_undirected.csv', 'w') as f:
        f.write('Source,Target,Weight\n')
        for u in G.adj:
            for v, w in G.adj[u].items():
                if u < v: # Avoid duplicates
                    f.write(f'{u},{v},{w}\n')

    # Export directed network
    with open('network_directed.csv', 'w') as f:
        f.write('Source,Target\n')
        for u in D.adj:
            for v in D.adj[u]:
                f.write(f'{u},{v}\n')

    print("Networks exported as CSV files")

```

```
# Create networks
G, D = create_networks()

# Export networks
export_for_cytoscape(G, D)

# Download files
files.download('network_undirected.csv')
files.download('network_directed.csv')

Networks exported as CSV files

<IPython.core.display.Javascript object>
<IPython.core.display.Javascript object>
<IPython.core.display.Javascript object>
<IPython.core.display.Javascript object>
```