Project Report: Network Analysis of Stock Price Data Introduction In this project, we perform a network analysis of stock price data. The objective is to explore the interconnections between different stocks based on their price movements over time. By constructing a network of stocks and analyzing its properties, we aim to gain insights into the underlying structure and dynamics of the stock market. Data Collection and Preprocessing We collected historical stock price data for a variety of stocks listed on the Indian stock market. The data was obtained from the National Stock Exchange (NSE) and consisted of daily closing prices for each stock over a specified time period. The data preprocessing steps included: Reversing the order of data to ensure chronological ordering. Converting closing prices to log returns to account for the natural logarithm of the ratio of successive closing prices. Handling missing values and formatting issues in the data. Network Construction We constructed a network representation of the stock market, where each stock is represented as a node in the network. The edges between nodes represent significant correlations in the log returns of the respective stocks. Correlation Calculation We calculated the Pearson correlation coefficient between the log returns of each pair of stocks. This coefficient measures the strength and direction of the linear relationship between the variables. Thresholding To create a sparse and meaningful network, we applied a threshold to the correlation matrix. Only correlations above a certain threshold value were considered as edges in the network. Adjacency Matrix The thresholded correlation matrix was converted into an adjacency matrix, where entries of 1 indicate the presence of an edge between two nodes (stocks), and entries of 0 indicate no edge. Network Analysis We conducted various analyses to explore the properties of the constructed network: Degree Centrality Degree centrality measures the number of connections (edges) a node has in the network. Nodes with higher degree centrality are considered more central in the network. Closeness Centrality Closeness centrality measures how close a node is to all other nodes in the network. Nodes with higher closeness centrality are more central in terms of their overall proximity to other nodes. Betweenness Centrality Betweenness centrality measures the extent to which a node lies on the shortest paths between other nodes in the network. Nodes with higher betweenness centrality play a crucial role in facilitating communication and information flow between other nodes. Visualization We visualized the constructed network using graph visualization techniques. Nodes represent individual stocks, and edges represent significant correlations between them. The size and color of nodes can be indicative of centrality measures such as degree, closeness, or betweenness centrality. Conclusion In conclusion, the network analysis of stock price data provides valuable insights into the underlying structure and dynamics of the stock market. By identifying key players (stocks) and understanding their relationships, investors and analysts can make more informed decisions regarding portfolio management, risk assessment, and market trends. Further research and analysis could involve exploring dynamic network properties, community detection, and predictive modeling based on network structure and dynamics. Report: Network Analysis of the Indian Stock Market Introduction: The provided code performs a network analysis on a dataset of Indian stock market data. The goal is to analyze the relationships and structure of the stock market by constructing a network based on the correlations between stock return prices. The code utilizes various libraries such as pandas, numpy, matplotlib, seaborn, and networkx to process the data, calculate correlations, and visualize the network. Data Preprocessing: The code starts by importing the necessary libraries and specifying the file paths for the dataset and modified dataset. It reads the list of files in the dataset directory, sorts them, and removes the first file (which is likely an index or metadata file). The modify function is defined to preprocess each stock data file. It reads the CSV file, reverses the order of rows, and extracts the 'close' column values. It converts the 'close' values to float and calculates the log return price using the 'CLOSE' column. The modified DataFrame is then saved to a new CSV file, and the return price is plotted over time for each stock. The modify function is applied to each file in the dataset directory, preprocessing all the stock data files. Correlation Calculation: The code then calculates the correlation matrix between the return prices of different stocks. It defines the build_data_matrix function, which reads each modified CSV file, extracts the 'RETURN_PRICE' column, and builds a data matrix where each column represents a different stock. The build_data_matrix function is applied to each file, constructing the complete data matrix. The correlation matrix is then calculated using the Pearson correlation coefficient method and printed. Network Construction: The code converts the correlation matrix to an adjacency matrix using a threshold value of 0.25. Correlations above the threshold are assigned a value of 1, indicating an edge between the corresponding stocks, while correlations below the threshold are assigned a value of 0. A graph is created using the networkx library, where nodes represent stocks, and edges represent correlations above a threshold of 0.5. The graph is visualized using a random layout, and the adjacency matrix is plotted as a heatmap using seaborn. Network Analysis: The code calculates various centrality measures for each node in the graph, including degree centrality, closeness centrality, and betweenness centrality. These measures provide insights into the importance and influence of each stock in the network. The centrality scores are printed for each node, and bar plots are created to visualize the distribution of centrality scores across the nodes. Conclusion: The code performs a comprehensive network analysis of the Indian stock market dataset. It preprocesses the stock data, calculates correlations between stock return prices, constructs a network based on the correlations, and analyzes the network using various centrality measures. The analysis provides insights into the structure and relationships of the stock market. The centrality measures help identify influential stocks and their roles within the network. The visualizations, such as the heatmap of the adjacency matrix and the bar plots of centrality scores, aid in understanding the network properties. Overall, the code demonstrates the application of network science techniques to analyze financial market data and gain valuable insights into the stock market dynamics.