Network Properties with Apache Spark:
Using pySpark and GraphFrames

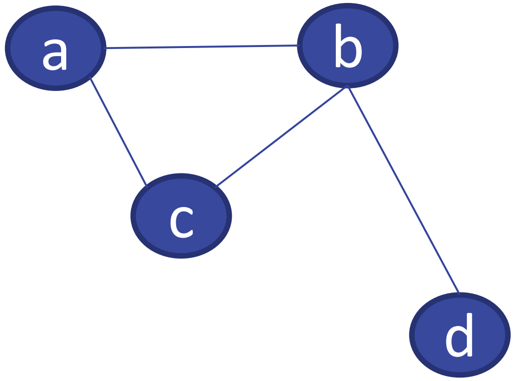
- Node degree distribution
- Node centrality
- Articulation points

Network Properties with Apache Spark with GraphFrames

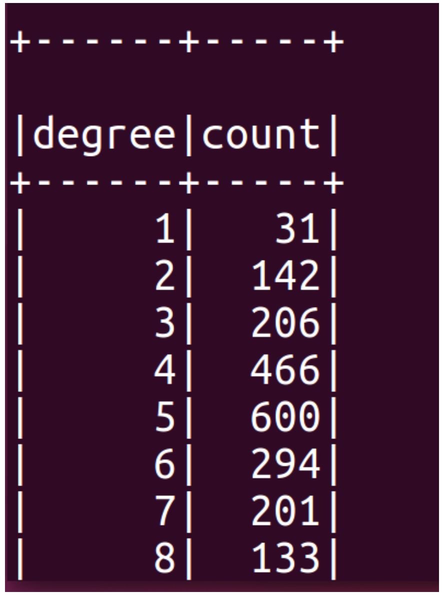
- **Step 1. Read edgelist from file and creating GraphFrame**
  - Extract pairs from input file and convert to data frame matching schema for graphframe edges
  - Extract all endpoints from input file to convert to dataframe matching scheme for graphFrame Vertices
- **Step 2. Calculate degree distribution of vertices**
  - Measure the frequency of nodes that have a certain degree value.
- **Step 3. Measure centrality of vertices**
  - Finding the distance between a vertex and all the other vertices
- **Step 4. Find articulation points**
  - Finding Cut Vertices

Step 2: Calculate Degree Distribution

- **Degree :**
  - The number of edges incident to the vertex
  - Example –
    - a,c=2; b= 3; d=1



Output:

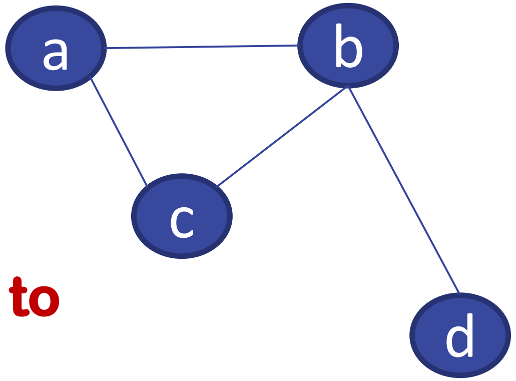


```
+------+-----+
|degree|count|
+------+-----+
|     1|   31|
|     2|  142|
|     3|  206|
|     4|  466|
|     5|  600|
|     6|  294|
|     7|  201|
|     8|  133|
```

Step 3: Measure Node Centrality

- **Closeness Centrality :**
  - Distance of a node to all other nodes

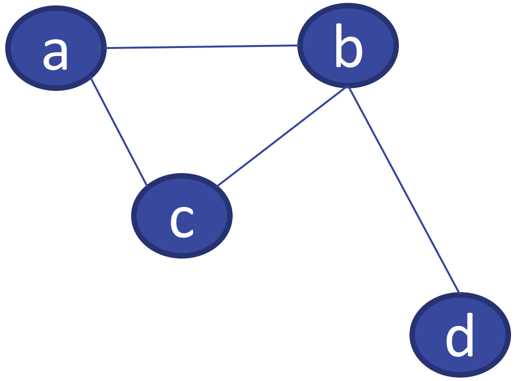
$$CC(v) = 1/\sum_{u \in V} d(u,v)$$



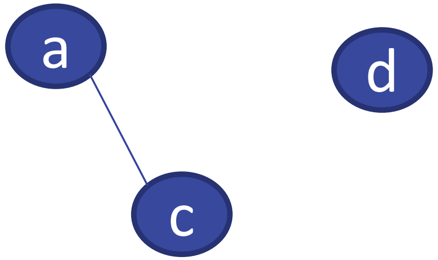
- **d(u,v) : Shortest path distance between u and v**
- **Measure of how long it takes for information in that node to spread to the other nodes**
- **Example :**
  - a = 1/(1 + 1 + 2) =1/4, b = 1/3, c = 1/4, d = 1/5

Step 4: Find Articulation Points (Cut Vertices)

- **Vertices that when removed create more disconnected components than there were originally in the network**



- **Example:**
  - Removing b creates two components



  - So, b is an **articulation point**
- **Critical to Communication**
  - Airline hubs
  - Traffic Routers
  - Power Energy Infrastructure