PROFEAT Tutorial

Biological Network Descriptor

Table of Contents

(A) Computational Flowchart.............................................................................................................. 1
(B) List of Network Descriptors.............................................................................................................. 2
(C) Sample Input & Output ...................................................................................................................... 9
  C.1 Overview of Input File Format ...................................................................................................... 9
  C.2 Overview of Output File Format .................................................................................................. 11
  C.3 Undirected Un-Weighted Network ............................................................................................... 12
  C.4 Undirected Edge-Weighted Network ............................................................................................. 13
  C.5 Undirected Node-Weighted Network............................................................................................. 14
  C.6 Undirected Edge-Node-Weighted Network .................................................................................... 15
  C.7 Directed Un-Weighted Network ................................................................................................... 16
  C.8 Multiple Networks in Single Input File ......................................................................................... 17
(D) Concepts and Algorithms of Network Descriptors ........................................................................ 19
  D.1 Node-Level Descriptors .............................................................................................................. 20
  D.2 Network-Level Descriptors ........................................................................................................ 29
  D.3 Edge-Level Descriptors .............................................................................................................. 46
(E) Computational Time Cost ............................................................................................................... 47
(F) Typical Applications of Network Descriptors in Systems Biology ............................................... 49
(G) Reference ........................................................................................................................................ 51

Prepared by: Zhang Peng (zhangpeng1202@gmail.com), 31 Dec 2016
(A) Computational Flowchart

Figure 1 | Computational flowchart for PROFEAT network descriptors
(B) List of Network Descriptors

Based on feature group indexing in PROFEAT, each network descriptor was indexed as $(X, Y, Z)$, where node-level descriptors were indexed by $X=G10$, network-level descriptors were indexed by $X=G11$, and edge-level descriptors were indexed by $X=G12$. Next, each descriptor was labelled as un-weighted, edge-weighted, node-weighted, or directed by $Y=1, 2, 3, 4$ respectively. The properties calculated by the normalized weight was labelled by an extra “$N$” in position of $Y$.

Lastly, $Z$ represented the descriptor ID# in the following Table 1, 2 and 3.

For example:

- $(G10, 1, 7)$: the node-level un-weighted neighbourhood connectivity
- $(G10, 2, 25)$: the node-level edge-weighted betweenness centrality
- $(G10, 4, 49)$: the node-level directed local clustering coefficient
- $(G11, 2N, 196)$: the network-level normalized edge-weighted transitivity
- $(G11, 3, 202)$: the network-level node-weighted global clustering coefficient
- $(G12, 2N, 2)$: the edge-level normalized edge-weighted edge betweenness

In the tables below, all descriptors were grouped into different categories according to their definitions and algorithms, and each column listed the computed descriptors for each network type. Some descriptors could be defined by either un-weighted connectivity information or weighted information. Therefore, some notations were given: “$\circ$” ($Y=1$) represents the features calculated based on un-weighted network adjacency, “$\rightarrow$” ($Y=2$) represents the features calculated based on edge weight, “$\bullet$” ($Y=3$) represents the features calculated based on node weight, and “$\ni$” ($Y=4$) represents the features calculated based on directed information.

Additionally, a slim set of network descriptors were selected, which is a cut-down version of the PROFEAT network descriptors that have been particularly applied in studying systems biology. The descriptors in slim set were marked by “$\bigstar$” in the ID column.
Table 1 | List of the node-level descriptors covered in PROFEAT

<table>
<thead>
<tr>
<th>ID</th>
<th>(G10) Node-Level Network Descriptor</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Un-Weighted</td>
</tr>
<tr>
<td>1</td>
<td>Degree</td>
<td>O</td>
</tr>
<tr>
<td>2</td>
<td>Scaled Connectivity</td>
<td>O</td>
</tr>
<tr>
<td>3</td>
<td>Number of Selfloops</td>
<td>O</td>
</tr>
<tr>
<td>4</td>
<td>Number of Triangles</td>
<td>O</td>
</tr>
<tr>
<td>5</td>
<td>Z Score</td>
<td>O</td>
</tr>
<tr>
<td>6</td>
<td>Clustering Coefficient</td>
<td>O</td>
</tr>
<tr>
<td>7</td>
<td>Neighborhood Connectivity</td>
<td>O</td>
</tr>
<tr>
<td>8</td>
<td>Topological Coefficient</td>
<td>O</td>
</tr>
<tr>
<td>9</td>
<td>Interconnectivity</td>
<td>O</td>
</tr>
<tr>
<td>10</td>
<td>Bridging Coefficient</td>
<td>O</td>
</tr>
<tr>
<td>11</td>
<td>Degree Centrality</td>
<td>O</td>
</tr>
<tr>
<td>12</td>
<td>Average Shortest Path Length</td>
<td>O</td>
</tr>
<tr>
<td>13</td>
<td>Distance Sum</td>
<td>O</td>
</tr>
<tr>
<td>14</td>
<td>Eccentricity</td>
<td>O</td>
</tr>
<tr>
<td>15</td>
<td>Eccentric</td>
<td>O</td>
</tr>
<tr>
<td>16</td>
<td>Deviation</td>
<td>O</td>
</tr>
<tr>
<td>17</td>
<td>Distance Deviation</td>
<td>O</td>
</tr>
<tr>
<td>18</td>
<td>Radiality</td>
<td>O</td>
</tr>
<tr>
<td>19</td>
<td>Closeness Centrality (avg)</td>
<td>O</td>
</tr>
<tr>
<td>20</td>
<td>Closeness Centrality (sum)</td>
<td>O</td>
</tr>
<tr>
<td>21</td>
<td>Eccentricity Centrality</td>
<td>O</td>
</tr>
<tr>
<td>22</td>
<td>Harmonic Closeness Centrality</td>
<td>O</td>
</tr>
<tr>
<td>23</td>
<td>Residual Closeness Centrality</td>
<td>O</td>
</tr>
<tr>
<td>24</td>
<td>Load Centrality</td>
<td>O</td>
</tr>
<tr>
<td>25</td>
<td>Betweenness Centrality</td>
<td>O</td>
</tr>
<tr>
<td>26</td>
<td>Normalized Betweenness</td>
<td>O</td>
</tr>
<tr>
<td>27</td>
<td>Bridging Centrality</td>
<td>O</td>
</tr>
<tr>
<td>28</td>
<td>CurrentFlow Betweenness</td>
<td>O</td>
</tr>
<tr>
<td>29</td>
<td>CurrentFlow Closeness</td>
<td>O</td>
</tr>
<tr>
<td>30</td>
<td>Eigenvector Centrality</td>
<td>O</td>
</tr>
<tr>
<td>31</td>
<td>Page Rank Centrality</td>
<td>O</td>
</tr>
<tr>
<td>32</td>
<td>Strength</td>
<td>—</td>
</tr>
<tr>
<td>33</td>
<td>Assortativity</td>
<td>—</td>
</tr>
<tr>
<td>34</td>
<td>Disparity</td>
<td>—</td>
</tr>
<tr>
<td>35</td>
<td>Geometric Mean of Triangles</td>
<td>—</td>
</tr>
<tr>
<td>36</td>
<td>Barrat's Local Clustering Coefficient</td>
<td>—</td>
</tr>
<tr>
<td>37</td>
<td>Onnela's Local Clustering Coefficient</td>
<td>—</td>
</tr>
<tr>
<td>38</td>
<td>Zhang's Local Clustering Coefficient</td>
<td>—</td>
</tr>
<tr>
<td>39</td>
<td>Holme's Local Clustering Coefficient</td>
<td>—</td>
</tr>
<tr>
<td>40</td>
<td>Edge-Weighted Interconnectivity</td>
<td>—</td>
</tr>
<tr>
<td>41</td>
<td>Node Weight</td>
<td>●</td>
</tr>
<tr>
<td>42</td>
<td>Node Weighted Cross Degree</td>
<td>●</td>
</tr>
<tr>
<td>43</td>
<td>Node Weighted Local Clustering Coeff.</td>
<td>●</td>
</tr>
<tr>
<td>44</td>
<td>Node-Weighted Neighbourhood Score</td>
<td>●</td>
</tr>
<tr>
<td>Directed Properties</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>★45 In-Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46 In-Degree Centrality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>★47 Out-Degree</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 Out-Degree Centrality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>★49 Directed Local Clustering Coefficient</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50 Neighbourhood Connectivity (only in)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51 Neighbourhood Connectivity (only out)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>52 Neighbourhood Connectivity (in &amp; out)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53 Average Directed Neighbour Degree</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 2** | List of the network-level descriptors covered in PROFEAT

<table>
<thead>
<tr>
<th>ID</th>
<th>(G11) Network-Level Network Descriptor</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Un-Directed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Un-Weighted</td>
</tr>
</tbody>
</table>

**Connectivity/Adjacency-based Properties**

| ★1  | Number of Nodes | O | O | O | O | O |
| ★2  | Number of Edges | O | O | O | O | O |
| ★3  | Number of Selfloops | O | O | O | O | O |
| ★4  | Maximum Connectivity | O | O | O | O | O |
| ★5  | Minimum Connectivity | O | O | O | O | O |
| ★6  | Average Number of Neighbours | O | O | O | O | O |
| 7   | Total Adjacency | O | O | O | O | O |
| ★8  | Network Density | O | O | O | O | O |
| ★9  | Average Clustering Coefficient | O | O | O | O | O |
| ★10 | Transitivity | O | O | O | O | O |
| ★11 | Heterogeneity | O | O | O | O | O |
| ★12 | Degree Centralization | O | O | O | O | O |
| 13  | Central Point Dominance | O | O | O | O | O |
| 14  | Degree Assortativity Coefficient | O | O | O | O | O |

**Shortest Path Length-based Properties**

| ★15 | Total Distance | O | O | O | O | O |
| ★16 | Network Diameter | O | O | O | O | O |
| ★17 | Network Radius | O | O | O | O | O |
| ★18 | Shape Coefficient | O | O | O | O | O |
| ★19 | Characteristic Path Length | O | O | O | O | O |
| 20  | Network Eccentricity | O | O | O | O | O |
| ★21 | Average Eccentricity | O | O | O | O | O |
| 22  | Network Eccentric | O | O | O | O | O |
| 23  | Eccentric Connectivity | O | O | O | O | O |
| 24  | Unipolarity | O | O | O | O | O |
| 25  | Integration | O | O | O | O | O |
| 26  | Variation | O | O | O | O | O |
| 27  | Average Distance | O | O | O | O | O |
| 28  | Mean Distance Deviation | O | O | O | O | O |
| 29  | Centralization | O | O | O | O | O |
| ★30 | Global Efficiency | O | O | O | O | O |

**Topological Indices**

<p>| 31  | Edge Complexity Index | O | O | O | O |
| ★32 | Randic Connectivity Index | O | O | O | O |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>Atom-Bond Connectivity Index</td>
</tr>
<tr>
<td>34</td>
<td>Zagreb Index 1</td>
</tr>
<tr>
<td>35</td>
<td>Zagreb Index 2</td>
</tr>
<tr>
<td>36</td>
<td>Zagreb Index Modified</td>
</tr>
<tr>
<td>37</td>
<td>Zagreb Index Augmented</td>
</tr>
<tr>
<td>38</td>
<td>Zagreb Index Variable</td>
</tr>
<tr>
<td>39</td>
<td>Narumi-Katayama Index</td>
</tr>
<tr>
<td>40</td>
<td>Narumi-Katayama Index (log)</td>
</tr>
<tr>
<td>41</td>
<td>Narumi Geometric Index</td>
</tr>
<tr>
<td>42</td>
<td>Narumi Harmonic Index</td>
</tr>
<tr>
<td>43</td>
<td>Alpha Index</td>
</tr>
<tr>
<td>44</td>
<td>Beta Index</td>
</tr>
<tr>
<td>45</td>
<td>Pi Index</td>
</tr>
<tr>
<td>46</td>
<td>Eta Index</td>
</tr>
<tr>
<td>47</td>
<td>Hierarchy</td>
</tr>
<tr>
<td>48</td>
<td>Robustness</td>
</tr>
<tr>
<td>49</td>
<td>Medium Articulation</td>
</tr>
<tr>
<td>50</td>
<td>Complexity Index A</td>
</tr>
<tr>
<td>51</td>
<td>Complexity Index B</td>
</tr>
<tr>
<td>52</td>
<td>Wiener Index</td>
</tr>
<tr>
<td>53</td>
<td>Hyper-Wiener</td>
</tr>
<tr>
<td>54</td>
<td>Harary Index 1</td>
</tr>
<tr>
<td>55</td>
<td>Harary Index 2</td>
</tr>
<tr>
<td>56</td>
<td>Compactness Index</td>
</tr>
<tr>
<td>57</td>
<td>Superpendentice Index</td>
</tr>
<tr>
<td>58</td>
<td>Hyper-Distance-Path Index</td>
</tr>
<tr>
<td>59</td>
<td>BalabanJ Index</td>
</tr>
<tr>
<td>60</td>
<td>BalabanJ-like 1 Index</td>
</tr>
<tr>
<td>61</td>
<td>BalabanJ-like 2 Index</td>
</tr>
<tr>
<td>62</td>
<td>BalabanJ-like 3 Index</td>
</tr>
<tr>
<td>63</td>
<td>Geometric Arithmetic Index 1</td>
</tr>
<tr>
<td>64</td>
<td>Geometric Arithmetic Index 2</td>
</tr>
<tr>
<td>65</td>
<td>Geometric Arithmetic Index 3</td>
</tr>
<tr>
<td>66</td>
<td>Szeged Index</td>
</tr>
<tr>
<td>67</td>
<td>Product Of Row Sums</td>
</tr>
<tr>
<td>68</td>
<td>Product Of Row Sums (log)</td>
</tr>
<tr>
<td>69</td>
<td>Schultz Topological Index</td>
</tr>
<tr>
<td>70</td>
<td>Gutman Topological Index</td>
</tr>
<tr>
<td>71</td>
<td>Efficiency Complexity</td>
</tr>
</tbody>
</table>

**Entropy-based Complexity Indices**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>Information Content (Degree Equality)</td>
</tr>
<tr>
<td>73</td>
<td>Information Content (Edge Equality)</td>
</tr>
<tr>
<td>74</td>
<td>Information Content (Edge Magnitude)</td>
</tr>
<tr>
<td>75</td>
<td>Information Content (Distance Degree)</td>
</tr>
<tr>
<td>76</td>
<td>Information Content (Distance Degree Equality)</td>
</tr>
<tr>
<td>77</td>
<td>Radial Centric Information Index</td>
</tr>
<tr>
<td>78</td>
<td>Distance Degree Compactness</td>
</tr>
<tr>
<td>79</td>
<td>Distance Degree Centric Index</td>
</tr>
<tr>
<td>80</td>
<td>Graph Distance Complexity</td>
</tr>
<tr>
<td>81</td>
<td>Information Layer Index</td>
</tr>
<tr>
<td>82</td>
<td>Bonchev Information Index 1</td>
</tr>
<tr>
<td>83</td>
<td>Bonchev Information Index 2</td>
</tr>
<tr>
<td>84</td>
<td>Bonchev Information Index 3</td>
</tr>
<tr>
<td>85</td>
<td>Balaban-like Information Index 1</td>
</tr>
<tr>
<td>86</td>
<td>Balaban-like Information Index 2</td>
</tr>
</tbody>
</table>
### Eigenvalue-based Complexity Indices

<table>
<thead>
<tr>
<th></th>
<th>Name</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>87</td>
<td>Graph Energy</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>88</td>
<td>Laplacian Energy</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>89</td>
<td>Spectral Radius</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>90</td>
<td>Estrada Index</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>91</td>
<td>Laplacian Estrada Index</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>92</td>
<td>Quasi-Weiner Index</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>93</td>
<td>Mohar Index 1</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>94</td>
<td>Mohar Index 2</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>95</td>
<td>Graph Index Complexity</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>96</td>
<td>Adjacency Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>97</td>
<td>Adjacency Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>98</td>
<td>Adjacency Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>99</td>
<td>Adjacency Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>100</td>
<td>Adjacency Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>101</td>
<td>Laplacian Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>102</td>
<td>Laplacian Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>103</td>
<td>Laplacian Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>104</td>
<td>Laplacian Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>105</td>
<td>Laplacian Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>106</td>
<td>Distance Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>107</td>
<td>Distance Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>108</td>
<td>Distance Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>109</td>
<td>Distance Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>110</td>
<td>Distance Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>111</td>
<td>Distance Path Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>112</td>
<td>Distance Path Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>113</td>
<td>Distance Path Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>114</td>
<td>Distance Path Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>115</td>
<td>Distance Path Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>116</td>
<td>Aug. Vertex Degree Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>117</td>
<td>Aug. Vertex Degree Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>118</td>
<td>Aug. Vertex Degree Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>119</td>
<td>Aug. Vertex Degree Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>120</td>
<td>Aug. Vertex Degree Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>121</td>
<td>Extended Adjacency Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>122</td>
<td>Extended Adjacency Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>123</td>
<td>Extended Adjacency Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>124</td>
<td>Extended Adjacency Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>125</td>
<td>Extended Adjacency Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>126</td>
<td>Vertex Connectivity Matrix HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>127</td>
<td>Vertex Connectivity Matrix SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>128</td>
<td>Vertex Connectivity Matrix ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>129</td>
<td>Vertex Connectivity Matrix PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>130</td>
<td>Vertex Connectivity Matrix IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>131</td>
<td>Random Walk Markov HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>132</td>
<td>Random Walk Markov SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>133</td>
<td>Random Walk Markov ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>134</td>
<td>Random Walk Markov PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>135</td>
<td>Random Walk Markov IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>136</td>
<td>Weighted Struct. Func. IM1 HM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>137</td>
<td>Weighted Struct. Func. IM1 SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>138</td>
<td>Weighted Struct. Func. IM1 ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>139</td>
<td>Weighted Struct. Func. IM1 PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>140</td>
<td>Weighted Struct. Func. IM1 IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>Weighted Struct. Func. IM2 HM (S=1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>----------------------------------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>141</td>
<td></td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>142</td>
<td>Weighted Struct. Func. IM2 SM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>143</td>
<td>Weighted Struct. Func. IM2 ISM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>144</td>
<td>Weighted Struct. Func. IM2 PM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>145</td>
<td>Weighted Struct. Func. IM2 IPM (S=1)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>146</td>
<td>Adjacency Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>147</td>
<td>Adjacency Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>148</td>
<td>Adjacency Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>149</td>
<td>Adjacency Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>150</td>
<td>Adjacency Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>151</td>
<td>Laplacian Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>152</td>
<td>Laplacian Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>153</td>
<td>Laplacian Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>154</td>
<td>Laplacian Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>155</td>
<td>Laplacian Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>156</td>
<td>Distance Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>157</td>
<td>Distance Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>158</td>
<td>Distance Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>159</td>
<td>Distance Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>160</td>
<td>Distance Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>161</td>
<td>Distance Path Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>162</td>
<td>Distance Path Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>163</td>
<td>Distance Path Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>164</td>
<td>Distance Path Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>165</td>
<td>Distance Path Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>166</td>
<td>Aug. Vertex Degree Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>167</td>
<td>Aug. Vertex Degree Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>168</td>
<td>Aug. Vertex Degree Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>169</td>
<td>Aug. Vertex Degree Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>170</td>
<td>Aug. Vertex Degree Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>171</td>
<td>Extended Adjacency Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>172</td>
<td>Extended Adjacency Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>173</td>
<td>Extended Adjacency Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>174</td>
<td>Extended Adjacency Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>175</td>
<td>Extended Adjacency Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>176</td>
<td>Vertex Connectivity Matrix HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>177</td>
<td>Vertex Connectivity Matrix SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>178</td>
<td>Vertex Connectivity Matrix ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>179</td>
<td>Vertex Connectivity Matrix PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>180</td>
<td>Vertex Connectivity Matrix IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>181</td>
<td>Random Walk Markov HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>182</td>
<td>Random Walk Markov SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>183</td>
<td>Random Walk Markov ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>184</td>
<td>Random Walk Markov PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>185</td>
<td>Random Walk Markov IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>186</td>
<td>Weighted Struct. Func. IM1 HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>187</td>
<td>Weighted Struct. Func. IM1 SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>188</td>
<td>Weighted Struct. Func. IM1 ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>189</td>
<td>Weighted Struct. Func. IM1 PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>190</td>
<td>Weighted Struct. Func. IM1 IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>191</td>
<td>Weighted Struct. Func. IM2 HM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>192</td>
<td>Weighted Struct. Func. IM2 SM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>193</td>
<td>Weighted Struct. Func. IM2 ISM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>194</td>
<td>Weighted Struct. Func. IM2 PM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>195</td>
<td>Weighted Struct. Func. IM2 IPM (S=2)</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>
### Edge-Weighted Properties

<table>
<thead>
<tr>
<th>ID</th>
<th>Property</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>196</td>
<td>Weighted Transitivity</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td>Barrat's Global Clustering Coefficient</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td>Onnela's Global Clustering Coefficient</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td>Zhang's Global Clustering Coefficient</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>Holme's Global Clustering Coefficient</td>
<td></td>
</tr>
</tbody>
</table>

### Node-Weighted Properties

<table>
<thead>
<tr>
<th>ID</th>
<th>Property</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>Total Node Weight</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>Node Weighted Global Clustering Coeff</td>
<td></td>
</tr>
</tbody>
</table>

### Directed Properties

<table>
<thead>
<tr>
<th>ID</th>
<th>Property</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>203</td>
<td>Average In-Degree</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td>Maximum In-Degree</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>Minimum In-Degree</td>
<td></td>
</tr>
<tr>
<td>206</td>
<td>Average Out-Degree</td>
<td></td>
</tr>
<tr>
<td>207</td>
<td>Maximum Out-Degree</td>
<td></td>
</tr>
<tr>
<td>208</td>
<td>Minimum Out-Degree</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td>Directed Global Clustering Coefficient</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td>Directed Flow Hierarchy</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3 | List of the edge-level descriptors covered in PROFEAT**

<table>
<thead>
<tr>
<th>ID</th>
<th>(G12) Edge-Level Network Descriptor</th>
<th>Network Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Un-Directed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Un-Weighted</td>
</tr>
<tr>
<td>1</td>
<td>Edge Weight</td>
<td>O</td>
</tr>
<tr>
<td>2</td>
<td>Edge-Betweenness</td>
<td>O</td>
</tr>
</tbody>
</table>
(C) Sample Input & Output

C.1 Overview of Input File Format
Currently, PROFEAT supports both SIF and NET network file format, where SIF is compatible with the majority of the network software (including Cytoscape 1, Gephi 2, GraphWeb 3, NAViGaTOR 4, PINA 5, SpectralNET 6), and NET format is used in Pajek 7.

SIF Network File Format
SIF, namely Simple Interaction File, is tab-delimited, specifying the two linked nodes in each line, with the relationship type in between. The following example illustrates the unweighted SIF file, where the biological binary interaction network could be protein-protein interaction network, gene regulatory network, gene co-expression network, drug-target network, etc.

\[ \text{Node A} \text{ tab } \text{Relationship} \text{ tab } \text{Node B} \]

Edge-weighted SIF is defined by extending the fourth column for numerical edge weight between the two connected nodes. In biological networks, the edge weight could be PPI kinetics constant, PPI binding affinity, gene co-expression association, interaction confidence level, etc.

\[ \text{Node A} \text{ tab } \text{Relationship} \text{ tab } \text{Node B} \text{ tab } \text{Edge Weight} \]

Directed SIF format is the same as the original SIF format, with the added direction information. For the two nodes in each line, the earlier one points to the latter one. Here, the example of unweighted SIF means that Node A points to Node B (A → B). Biologically, the directed network usually represents the oriented process map (e.g. signalling pathway, metabolic reaction, etc.).

NET Network File Format
NET format, developed by Pajek, mainly includes 3 sections (*vertices, *edges, and *arcs) in this file structure, where (1) *vertices section lists all the nodes; (2) *edges section lists all the undirected interactions between two nodes, with an optional edge weight in the third column; and (3) *arcs section lists all the directed interactions, pointing from the earlier node to the later node.
The above example means there are 3 nodes (*vertices) A, B, C in the network, where there is an undirected interaction (*edge) between A and C, and a directed interaction (*arcs) from B to C.

**TXT Node Weight File Format**

The node weight file is separated from the network file. It follows the tab-delimited txt format, specifying the node ID and its numerical node weight, while the node ID must be matched with the network file. Biologically, node weight represents the molecular level (e.g. gene expression).

\[
\text{[Node ID] tab [Node Weight]}
\]

**Table 4 | Required file(s) for each input network type**

<table>
<thead>
<tr>
<th>Input Network Type</th>
<th>Required File(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unweighted Network File</td>
</tr>
<tr>
<td>Undirected Un-Weighted Network</td>
<td>✓</td>
</tr>
<tr>
<td>Undirected Edge-Weighted Network</td>
<td></td>
</tr>
<tr>
<td>Undirected Node-Weighted Network</td>
<td>✓</td>
</tr>
<tr>
<td>Undirected Edge-Node-Weighted Network</td>
<td></td>
</tr>
<tr>
<td>Directed Un-Weighted Network</td>
<td></td>
</tr>
</tbody>
</table>
C.2 Overview of Output File Format

For an un-weighted input network, PROFEAT provides only the un-weighted descriptors. For a (edge, node, or both) weighted input network, PROFEAT will compute the un-weighted features, the original weighted features, and the normalized weighted features.

The output file is well organized in text format, by giving (1) a header started with “!” including the input network file name, the total number of networks, the total number of nodes, and the total number of edges; (2) the node-level descriptors; (3) the network-level descriptors; and (4) the edge-level descriptors respectively.

If there are multiple separated networks in the single input file, PROFEAT will automatically detect them, rank them, rename them, and compute the descriptors for each individual network accordingly. This function is embedded in all types of input networks. For such case study, please refer to the later section “Multiple Networks in Single Input File” for the details.

After submitting the job, a unique network id (net-x) and a URL will be given. Users could save the URL (e.g. http://bidd2.nus.edu.sg/cgi-bin/profeat2016/network/profeat-result.cgi?uid=net-x) for accessing the results later, in case that it may take some time to finish computing the large networks.
### C.3 Undirected Un-Weighted Network

#### Table 5 | Sample input & output of an undirected un-weighted network

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Network in SIF</th>
<th>Network in NET</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sample Input</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Network Graphics</strong></td>
<td><strong>Network in SIF</strong></td>
<td><strong>Network in NET</strong></td>
</tr>
<tr>
<td>A interact B</td>
<td>A B</td>
<td></td>
</tr>
<tr>
<td>B interact C</td>
<td>C D</td>
<td></td>
</tr>
<tr>
<td>B interact D</td>
<td>E F</td>
<td></td>
</tr>
<tr>
<td>B interact E</td>
<td>H I</td>
<td></td>
</tr>
<tr>
<td>C interact D</td>
<td>J K</td>
<td></td>
</tr>
<tr>
<td>D interact E</td>
<td>L M</td>
<td></td>
</tr>
<tr>
<td>C interact E</td>
<td>A B</td>
<td>vertices</td>
</tr>
<tr>
<td>A interact F</td>
<td>C D</td>
<td></td>
</tr>
<tr>
<td>L interact F</td>
<td>E F</td>
<td></td>
</tr>
<tr>
<td>L interact K</td>
<td>G H</td>
<td></td>
</tr>
<tr>
<td>K interact K</td>
<td>I J</td>
<td></td>
</tr>
<tr>
<td>L interact J</td>
<td>K L</td>
<td></td>
</tr>
<tr>
<td>L interact I</td>
<td>L M</td>
<td></td>
</tr>
<tr>
<td>L interact H</td>
<td>G H</td>
<td></td>
</tr>
<tr>
<td>L interact G</td>
<td>J K</td>
<td></td>
</tr>
</tbody>
</table>

#### Sample Output

```
! Input Network File Name: sample_network.sif
! Total Number of Networks: 1
! Total Number of Nodes: 12
! Total Number of Edges: 15
! PROFET Network Descriptor: Slim Set

# Network File: sample_network.sif {12 Nodes; 15 Edges}
# # Node-level Descriptors
[GI0.0.0] Node Label: A B ... L
[GI0.1] Un-Weighted Features
[GI0.1.1] Degree: 2 4 ... 6

# Network-Level Descriptors
[GI1.1] Un-Weighted Features
[GI1.1.1] Number of Nodes: 12
[GI1.1.2] Number of Edges: 15

# Edge-level Descriptors

<table>
<thead>
<tr>
<th>NodeLabel 1</th>
<th>NodeLabel 2</th>
<th>Edge Betweenness</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>0.49</td>
</tr>
</tbody>
</table>
```

---

12
### C.4 Undirected Edge-Weighted Network

Table 6 | Sample input & output of an undirected edge-weighted network

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Network Graphics</th>
<th>Network in SIF</th>
<th>Network in NET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Sample Output

```plaintext
! Input Network File Name: sample_network_edge_weighted.sif
! Total Number of Networks: 1
! Total Number of Nodes: 12
! Total Number of Edges: 15
! PROFEAT Network Descriptor: Slim Set

# Network Files: sample_network_edge_weighted.sif (12 Nodes; 15 Edges)
# # Node-Level Descriptors
[G10.0.0]  Node Label: A B ... L
[G10.1]    Un-Weighted Features
[G10.1.1]  Degree: 2 4 ... 6
...        ...        ...
[G10.2]    Original Edge-Weighted Features
[G10.2.11]  Edge-Weight Avg Shortest Path Length: 1.06 1.24 ... 1.0
...        ...        ...
[G10.2N]   Normalized Edge-Weighted Features
[G10.2N.11] N. Edge-Weight Avg Shortest Path Length: 0.46 0.56 ... 0.46
...        ...        ...
# # Network-Level Descriptors
[G11.1]    Un-Weighted Features
[G11.1.1]  Number of Nodes: 12
[G11.1.2]  Number of Edges: 15
...        ...        ...
[G11.2]    Original Edge-Weighted Features
[G11.2.14]  Edge-Weight Total Distance: 93.0
...        ...        ...
[G11.2N]   Normalized Edge-Weighted Features
[G11.2N.14] N. Edge-Weight Total Distance: 45.2
...        ...        ...
# # Edge-Level Descriptors
NodeLabel 1  NodeLabel 2  Edge Weight  ...  Edge Betweenness
A            B            2                          0.455
...        ...        ...        ...
```

Edge Weight

![Diagram of an undirected edge-weighted network](image-url)
# C.5 Undirected Node-Weighted Network

Table 7 | Sample input & output of an undirected node-weighted network

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Network Graphics</th>
<th>Network in SIF</th>
<th>Network in NET</th>
<th>Node Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="Diagram" /></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Sample Output

```plaintext
! Input Network File Name: sample_network.sif
! Input Node Weight File Name: sample_network_node_weighted.txt
! Total Number of Networks: 1
! Total Number of Nodes: 12
! Total Number of Edges: 15
! PROFEAT Network Descriptor: Slim Set

# Network File: sample_network.sif [12 Nodes; 15 Edges]

# # Node-Level Descriptors
[G10.0.0] Node Label: A B ... L
[G10.1] Un-Weighted Features
[N10.1.1] Degree: 2 4 ... 6
[N10.3] Original Node-Weighted Features
[N10.3.38] Node Weight: 3 2 ... 3
[N10.3N] Normalized Node-Weighted Features
[N10.3N.35] N. Node Weight: 1 0.5 ... 1

# # Network-Level Descriptors
[G11.1] Un-Weighted Features
[G11.1.1] Number of Nodes: 12
[G11.1.2] Number of Edges: 15
[N11.3] Original Node-Weighted Features
[N11.3.150] Total Node Weight: 23
[N11.3N] Normalized Node-Weighted Features
[N11.3N.150] N. Total Node Weight: 5.53

# # Edges-Level Descriptors

<table>
<thead>
<tr>
<th>NodeLabel 1</th>
<th>NodeLabel 2</th>
<th>Edge Betweenness</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>0.49</td>
</tr>
</tbody>
</table>
```
### C.6 Undirected Edge-Node-Weighted Network

#### Table 8 | Sample input & output of an undirected edge-node-weighted network

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Network Graphics</strong></td>
<td><strong>Network in SIF</strong></td>
</tr>
<tr>
<td><img src="image" alt="Diagram" /></td>
<td>A interact B 2</td>
</tr>
<tr>
<td></td>
<td>B interact C 3</td>
</tr>
<tr>
<td></td>
<td>B interact D 2</td>
</tr>
<tr>
<td></td>
<td>B interact E 3</td>
</tr>
<tr>
<td></td>
<td>C interact D 1</td>
</tr>
<tr>
<td></td>
<td>D interact E 2</td>
</tr>
<tr>
<td></td>
<td>C interact E 1</td>
</tr>
<tr>
<td></td>
<td>A interact F 3</td>
</tr>
<tr>
<td></td>
<td>L interact F 2</td>
</tr>
<tr>
<td></td>
<td>L interact K 3</td>
</tr>
<tr>
<td></td>
<td>K interact J 1</td>
</tr>
<tr>
<td></td>
<td>L interact I 2</td>
</tr>
<tr>
<td></td>
<td>L interact H 1</td>
</tr>
<tr>
<td></td>
<td>L interact G 1</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

```
# Network File: sample_network_edge_weighted.sif {12 Nodes; 15 Edges}
# # Node-Level Descriptors
# [G10.0.0] Node Label: A B E ... L
# [G10.1] Un-Weighted Features
# [G10.1.1] Degree: 2 4 ... 5
# [G10.2] Original Edge-Weighted Features
# [G10.3] Original Node-Weighted Features
# [G10.3N] Normalized Edge-Weighted Features
# [G10.3N] Normalized Node-Weighted Features
# # # Network-Level Descriptors
# [G11.1] Un-Weighted Features
# [G11.1.1] Number of Nodes: 12
# [G11.1.2] Number of Edges: 15
# [G11.2] Original Edge-Weighted Features
# [G11.3] Original Node-Weighted Features
# [G11.3N] Normalized Edge-Weighted Features
# [G11.3N] Normalized Node-Weighted Features
# # # Edge-Level Descriptors
# [G12.1] Edge Weight
# [G12.1.1] Edge Betweenness
# A ... B 2 0.485
```
### C.7 Directed Un-Weighted Network

#### Table 9 | Sample input & output of an directed un-weighted network

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Network in SIF</th>
<th>Network in NET</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Network Graphics</strong></td>
<td><strong>Network in SIF</strong></td>
<td><strong>Network in NET</strong></td>
</tr>
<tr>
<td><img src="image1.png" alt="Network Graphics" /></td>
<td>A direct-to B</td>
<td><strong>vertices</strong></td>
</tr>
<tr>
<td></td>
<td>C direct-to B</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>D direct-to B</td>
<td>B</td>
</tr>
<tr>
<td></td>
<td>E direct-to B</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>D direct-to C</td>
<td>D</td>
</tr>
<tr>
<td></td>
<td>E direct-to D</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>E direct-to C</td>
<td>F</td>
</tr>
<tr>
<td></td>
<td>F direct-to A</td>
<td>G</td>
</tr>
<tr>
<td></td>
<td>F direct-to L</td>
<td>H</td>
</tr>
<tr>
<td></td>
<td>K direct-to L</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>J direct-to L</td>
<td>J</td>
</tr>
<tr>
<td></td>
<td>L direct-to I</td>
<td>K</td>
</tr>
<tr>
<td></td>
<td>L direct-to H</td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>L direct-to G</td>
<td>L</td>
</tr>
</tbody>
</table>

**Sample Output**

```plaintext
! Input Network File Name: sample_network_directed.sif
! Total Number of Networks: 1
! Total Number of Nodes: 12
! Total Number of Edges: 15
! PROFEAT Network Descriptor: Slim Set

# Network File: sample_network_directed.sif {12 Nodes; 15 Edges}

```

```plaintext
# # Node-Level Descriptors
[G10.0.0] Node Label:  A  B  ...  L
[G10.4] Directed Features
[G10.4.1] In-Degree:  1  4  ...  3
[G10.4.2] Out-Degree:  1  0  ...  3
...
```

```plaintext
# # Network-Level Descriptors
[G11.4] Directed Features
[G11.4.1] Number of Nodes:  12
[G11.4.2] Number of Edges:  15
...
```

```plaintext
```

```plaintext
```
C.8 Multiple Networks in Single Input File

Quantitative network analysis may get trouble by having mixed networks in data collection. The available tools have not yet provided the function to split the disconnected network from a single input. To illustrate this function, “sample_network_multiple.sif”, containing 3 separated networks, is inputted. PROFEAT analyses the global adjacency, splits the input into 3 new files, ranks by the number of nodes, and renames by adding suffix. Finally, each network is proceed for descriptor calculation accordingly.

Table 10 | Sample input & output of a single file containing disconnected networks

<table>
<thead>
<tr>
<th>Sample Input</th>
<th>Network in SIF</th>
<th>Network in NET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Graphics</td>
<td>Network in SIF</td>
<td>Network in NET</td>
</tr>
<tr>
<td></td>
<td>vertices</td>
<td>edges</td>
</tr>
<tr>
<td>D interact E</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>D interact F</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>D interact G</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>D interact H</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>D interact I</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>D interact J</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>D interact K</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td>D interact L</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>D interact M</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td>D interact N</td>
<td>J</td>
<td>J</td>
</tr>
<tr>
<td>D interact K</td>
<td>K</td>
<td>K</td>
</tr>
<tr>
<td>D interact L</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>D interact M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>D interact N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>D interact K</td>
<td>A</td>
<td>A</td>
</tr>
<tr>
<td>D interact L</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>D interact M</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>D interact N</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>D interact K</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>D interact L</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>D interact M</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td>D interact N</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>D interact K</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td>D interact L</td>
<td>J</td>
<td>J</td>
</tr>
<tr>
<td>D interact M</td>
<td>K</td>
<td>K</td>
</tr>
<tr>
<td>D interact N</td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>D interact K</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>D interact L</td>
<td>I</td>
<td>I</td>
</tr>
<tr>
<td>D interact M</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td>D interact N</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>
Sample Output

! Input Network File Name: sample_network_multiple.sif
! Total Number of Networks: 3
! Total Number of Nodes: 14
! Total Number of Edges: 15
! PROFEAT Network Descriptor: Slim Set

# Network File: sample_network_multiple_sub_1.sif {7 Nodes; 7 Edges}
# # Node-Level Descriptors
[G10.0.0] Node Label: F G ... L
[G10.1] Un-Weighted Features
... ... ... ... ...
# # Network-Level Descriptors
[G11.1] Un-Weighted Features
... ... ...
# # Edge-Level Descriptors
... ... ...

# Network File: sample_network_multiple_sub_2.sif {4 Nodes; 6 Edges}
# # Node-Level Descriptors
[G10.0.0] Node Label: B C D E
[G10.1] Un-Weighted Features
... ... ... ... ...
# # Network-Level Descriptors
[G11.1] Un-Weighted Features
... ... ...
# # Edge-Level Descriptors
... ... ...

# Network File: sample_network_multiple_sub_3.sif {3 Nodes; 2 Edges}
# # Node-Level Descriptors
[G10.0.0] Node Label: A M N
[G10.1] Un-Weighted Features
... ... ... ... ...
# # Network-Level Descriptors
[G11.1] Un-Weighted Features
... ... ...
# # Edge-Level Descriptors
... ... ...
(D) Concepts and Algorithms of Network Descriptors

For a connected and undirected network, some basic information matrices will be generated:

1. Un-weighted matrix
   1.1. Adjacency matrix “A”, with $A_{ij}=A_{ji}=1$, if exists an edge linking node $i$ and node $j$.
      Otherwise, $A_{ij}=A_{ji}=0$.

2. Edge-weight matrix
   2.1. Edge weight matrix “$EW_i$”, assigning $EW_{ij}=EW_{ji}$ edge weight between node $i$ and $j$.
   2.2. Normalized edge weight matrix “Nor$EW_{ij}$”, defined as below. Here, the constant factor 0.99 in the denominator is to slightly enlarge the domain from the minimum value to the maximum value, ensuring the normalized minimum edge weight will not be zero.

\[
NorEW_{ij} = \frac{EW_{ij} - \min\{EW\}}{\max\{EW\} - 0.99 \times \min\{EW\}}
\]

3. Node-weighted matrix
   3.1. Node weight list “$NW_i$”, where $NW_i$ = node weight of node $i$, based on the input data.
   3.2. Normalized node weight list “Nor$NW_i$”. Again, the constant 0.99 in the denominator is to ensure the normalized minimum node weight will not be zero.

\[
NorNW_i = \frac{NW_i - \min\{NW\}}{\max\{NW\} - 0.99 \times \min\{NW\}}
\]

For a connected and directed network, directed adjacency matrix will be generated:

4. Un-weighted matrix
   4.1. Directed adjacency matrix “$a$”, where $a_{ij}=1$, if exists a directed link from node $i$ pointing to node $j$. $a_{ji}=1$ only if exists another directed link from node $j$ pointing to node $i$.

The network descriptors will be introduced according to their order in Table 1, 2, and 3 given previously. As some descriptors can be derived from either un-weighted adjacency matrix or weighted matrix, we will mainly introduce the un-weighted ones, and the weighted ones can be easily obtained by substituting the algorithm with the weighted matrix.
D.1 Node-Level Descriptors

Feature Category: Connectivity/Adjacency-based Properties

1. Degree

Degree of a node $i$ "deg,$i$" is the number of edges linked to it.

2. Scaled Connectivity

$$scaledConnect_i = \frac{deg_i}{\max\{deg\}}$$

3. Number of Selfloops

Selfloops of a node $i$ "selfloop,$i$" is the number of edges linking to itself.

4. Number of Triangles $^8$

$$tri_i = \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N A_{ij}A_{ik}A_{jk}$$

5. Z Score $^9,10$

Z score is a connectivity index of a node, based on the degree distribution of a network. It has been applied in discovering network motifs in some studies.

$$zscore_i = \frac{deg_i - \text{avg}(deg)}{\text{dev}(deg)}$$

6. Clustering Coefficient $^{11,12}$

The clustering coefficient of a node $i$ is defined as below, where $e_i$ is the number of connected pairs between all neighbours of node $i$. It is assumed to be 0, if less than two neighbours.

$$cluster_i = \frac{2e_i}{deg_i(deg_i - 1)}$$

7. Neighborhood Connectivity $^{13}$

The connectivity of a node is the number of its neighbours. The neighbourhood connectivity of a node $i$ is defined as its average connectivity of all neighbours.

$$neighbourConnect_i = \frac{\sum_{j=1}^N A_{ij} \, deg_j }{deg_i}$$

8. Topological Coefficient $^{14}$

In calculating topological coefficient, $j$ represents all the nodes sharing at least one neighbour with $i$, and $J(i, j)$ is the number of shared neighbours between $i$ and $j$. If there is a direct edge between $i$ and $j$, plus an additional 1 to $J(i, j)$. It is a measure to estimate the tendency of the nodes to share neighbours.

$$\text{topology}_i = \text{avg}(J(i,j)/deg_i)$$
9. **Interconnectivity**\(^{15,16,17}\)

Firstly, the interconnectivity score is generated for each edge in the network. \(N(i)\) is the neighbours of node \(i\), such that \(|N(i) \cap N(j)|\) is the number of shared neighbours between node \(i\) and node \(j\).

\[
ICN_{\text{edge}}_{ij} = A_{ij} \left( \frac{2 + |N(i) \cap N(j)|}{\sqrt{\text{deg}_i \cdot \text{deg}_j}} \right)
\]

Next, the interconnectivity for each node is calculated based on the \(ICN_{\text{edge}}\) scores.

\[
ICN_{\text{node}}_i = \frac{1}{\text{deg}_i} \sum_{j=1}^{N} ICN_{\text{edge}}_{ij}
\]

10. **Bridging Coefficient**\(^{18}\)

The bridging coefficient describes how well the node is linked between high-degree nodes.

\[
bridge_i = \frac{\text{deg}_i^{-1}}{\sum_{j=1}^{N} A_{ij} \cdot \frac{1}{\text{deg}_j}}\]

11. **Degree Centrality**\(^{19}\)

\[
\text{centralityDeg}_i = \frac{\text{deg}_i}{N - 1}
\]

**Feature Category: Shortest Path Length-based Properties**

12. **Average Shortest Path Length**\(^{20}\)

Shortest path lengths are computed by Dijkstra’s algorithm to generate an \(N \times N\) matrix for storing the pairwise shortest path lengths, such that \(D_{ij}\) is the shortest path length between node \(i\) and node \(j\). For an unweighted network, the shortest path length is basically the minimum number of edges linking between any two nodes. For an edge-weighted network, the weighted shortest path length could be generated based on the edge weight matrix. Here, \(avgSPL_i\) is the average length of shortest paths between node \(i\) and all other nodes.

\[
avgSPL_i = \frac{1}{N - 1} \sum_{j=1}^{N} D_{ij}
\]

13. **Distance Sum**\(^{21}\)

Distance sum is obtained by adding up all the shortest paths from node \(i\).

\[
distSum_i = \sum_{j=1}^{N} D_{ij}
\]
14. Eccentricity

Eccentricity is the maximum non-infinite shortest path length between node \( i \) and all the other nodes.

\[
\text{eccentricity}_i = \max \{ D_{ij} \}
\]

15. Eccentric

Different from eccentricity measure, eccentric index is the absolute difference between the nodes’ eccentricities and the graph’s average eccentricity.

\[
\text{eccentric}_i = |\text{eccentricity}_i - \text{avg} \{ \text{eccentricity}_G \}|
\]

16. Deviation

Node’s deviation measures the difference between the node’s distance sum and the graph’s unipolarity, where the unipolarity is defined as the minimum of distance sums among all nodes.

\[
\text{deviation}_i = \text{distSum}_i - \text{unipolarity}_G
\]

17. Distance Deviation

This is the absolute difference between nodes’ distance sum and graph’s average distance.

\[
\text{distDev}_i = |\text{distSum}_i - \text{distAvg}_G|
\]

18. Radiality

Radiality is computed by subtracting the average shortest path length of node \( i \) from the diameter plus 1, and the result is then divided by the network diameter.

High value of radiality implies the node is generally nearer to other nodes, while a low radiality indicates the node is peripheral in the network.

\[
\text{radiality}_i = \frac{\text{diameter}_G - \text{avgSPL}_i + 1}{\text{diameter}_G}
\]

19. Closeness Centrality (avg)

The closeness centrality of a node is defined as the reciprocal of the average shortest path length. It measures how fast information spreads from a given node to other reachable nodes in the network.

\[
\text{centralityCloseAvg}_i = \frac{1}{\frac{1}{N} \sum_{j=1}^{N} D_{ij}}
\]

20. Closeness Centrality (sum)

\[
\text{centralityCloseSum}_i = \frac{1}{\sum_{j=1}^{N} D_{ij}}
\]

21. Eccentricity Centrality

\[
\text{centralityEccentricity}_i = \frac{1}{\max \{ D_{ij} \}}
\]
22. Harmonic Centrality

The harmonic closeness is the sum of reciprocals of average shortest path lengths for each node.

\[ \text{centralityHar}_i = \sum_{j=1}^{N} \frac{1}{D_{ij}} \]

23. Residual Centrality

\[ \text{centralityRes}_i = \sum_{j=1}^{N} \frac{1}{2D_{ij}} \]

24. Load Centrality

The load centrality of a node \( i \) is the fraction of all shortest paths that passing through the node \( i \). A node has a high load centrality if it is involved in a high number of shortest paths.

25. Betweenness Centrality

The betweenness centrality quantifies the number of times a node serving as a linking bridge along the shortest path between two other nodes. It is computed by the following equation, where \( s \) and \( t \) are the nodes different from \( i \) in the network, \( \sigma_{st}(i) \) is the number of shortest paths from \( s \) to \( t \) that passing through \( i \), and \( \sigma_{st} \) is the number of shortest paths from \( s \) to \( t \). The betweenness centrality reflects the extent of control of that node exerting over the interactions with other nodes in the network.

\[ \text{centralityBtw}_i = \frac{\sum_{\text{exists}} \sigma_{st}(i)}{\sigma_{st}} \]

26. Normalized Betweenness Centrality

\[ \text{centralityBtwNor}_i = \frac{\text{centralityBtw}_i - \min{\text{centralityBtw}_G}}{\max{\text{centralityBtw}_G} - \min{\text{centralityBtw}_G}} \]

27. Bridging Centrality

The bridging centrality of a node is the product of the bridging coefficient and the betweenness centrality. A higher bridging centrality means more information flowing through that node.

\[ \text{centralityBridge}_i = \text{bridge}_i \cdot \text{centralityBtw}_i \]

28. Current Flow Betweenness

Previously, the betweenness centrality is based on the shortest path length in the network. Here, the current flow betweenness centrality is assumed that information efficiently spreading in the network like an electrical current, as a current flow analog.

Firstly, the resistance \( R \) of an edge is defined, where \( r(e) = 1/w(e) \) and \( w(e) \) is the weight of an edge \( e \). For unweighted networks, \( w(e) = 1 \) for all edges.

Secondarily, a vector \( b \), namely supply, is defined where current enters and leaves the network. Since there should be as much current entering as leaving the network, \( \sum b(v) = 0 \).
Thirdly, the electrical current $c$ is defined and it should follow the law below.

**Kirchhoff’s Current Law** (for every $v \in V$):
\[
\sum_{(v,w) \in E} c(v,w) - \sum_{(u,v) \in E} c(u,v) = b(v)
\]

**Kirchhoff’s Potential Law** (for every current cycle $e_1, \ldots, e_k$ in the network):
\[
\sum_{i=1}^{k} c(e_i) = 0
\]

Lastly, the potential difference $p$ is defined by **Ohm’s Law**, where $p(e)=c(e)/r(e)$. To calculate the current flow betweenness, throughput $\tau(v)$ of a node $v$, and throughput $\tau(e)$ of an edge $e$ are defined:
\[
\tau(v) = \frac{1}{2} \left( -|b(v)| + \sum_{e} |c(e)| \right)
\]
\[
\tau(e) = |c(e)|
\]

Therefore, current flow betweenness (sometimes also called random-walk betweenness) is then defined, where $\tau_{st}$ denotes the throughput of a $s$-$t$ current, and $N_b = (N-1)(N-2)$.

\[
CF\text{between}_i = \frac{1}{N_b} \sum_{s,t \in V} \tau_{st}(i)
\]

**29. Current Flow Closeness**

The current flow closeness centrality is a variant of the current flow betweenness centrality, by using the analog of shortest path length in electrical networks.

\[
CF\text{close}_i = \frac{N_c}{\sum_{s \neq t} p_{st}(s) - p_{st}(t)}
\]

Where, $N_c = (N-1)$, and $p_{st}(s)-p_{st}(t)$ denotes the effective resistance of $s$-$t$ current, interpreted as an alternative measure of distance between node $s$ and node $t$. 
**Feature Category: Eigenvector-based Centrality Indices**

30. Eigenvector Centrality \(^{32,33}\)

Eigenvector centrality is the eigenvalue-based methods to approximate the importance of each node in a network. It assumes that each node's centrality is the sum of its neighbors’ centrality values, which is saying that an important node should be linking to important neighbors.

In algorithm, the eigenvector centralities for all nodes are initialized to 1 at the beginning, and then an eigenvalue-based function is applied to iteratively converge the centrality to a fixed value, by considering the neighbourhood relationships and the neighbors’ centrality values. Let \(\{\lambda_1, \lambda_2 \ldots \lambda_k\}\) be the non-zero eigenvalues of adjacency matrix of the network, and \(\lambda_{\text{max}}\) is the maximum eigenvalue.

\[
\text{centralityEigen}_i = \frac{1}{\lambda_{\text{max}}} \sum_{j=1}^{N} A_{ij} \cdot \text{centralityEigen}_j
\]

31. PageRank Centrality \(^{34,35,36,37,38,39}\)

PageRank is an algorithm implemented in Google search engine to rank the websites, according to the webpage connections in the World Wide Web. It is a variant of eigenvector centrality, by initializing the PageRank centralities to an equal probability value \(1/N\) for all nodes.

The equation will iteratively update the node centrality value by using a constant damping factor \(d\), its neighbors’ PageRank centrality value, and its degree. The algorithm stops running, when the PageRank centrality converges, and the damping factor \(d\) is generally assumed to 0.85.

\[
\text{pageRank}_i = \frac{1 - d}{N} + d \cdot \sum_{j=1}^{N} A_{ij} \cdot \frac{\text{pageRank}_j}{\text{deg}_j}
\]

**Feature Category: Edge-Weighted Properties**

32. Strength \(^{40}\)

The strength for each vertex is defined as the sum of all the edge weights connected to that vertex.

\[
\text{strength}_i = \sum_{j=1}^{N} A_{ij} \cdot W_{ij}
\]

33. Assortativity \(^{40,41}\)

In an unweighted graph, assortativity is as the same as the previously defined neighbourhood connectivity. For a weighted graph, it is defined as below.

\[
\text{assortativity}_i = \frac{1}{\text{strength}_i} \sum_{j=1}^{N} W_{ij} \cdot \text{deg}_j
\]
34. Disparity
\[ \text{disparity}_i = \sum_{j=1}^{N} \left( A_{ij} \cdot W_{ij} \right)^2 \]

35. Geometric Mean of Triangles
\[ \text{geo\_tri}_i = \frac{1}{2} \sum_{j=1}^{N} \sum_{k=1}^{N} \sqrt{W_{ij}W_{ik}W_{jk}} \]

36. Barrat’s Local Clustering Coefficients
\[ \text{clusterBarrat}_i = \frac{1}{\text{strength}_i (\text{deg}_i - 1)} \sum_{j=1}^{N} \sum_{k=1}^{N} \left( A_{ij}A_{ik}A_{jk} \cdot \frac{W_{ij} + W_{ik}}{2} \right) \]

37. Onnela’s Local Clustering Coefficients
\[ \text{clusterOnnela}_i = \frac{1}{\text{deg}_i \cdot (\text{deg}_i - 1)} \sum_{j=1}^{N} \sum_{k=1}^{N} \left( \tilde{W}_{ij}\tilde{W}_{ik}\tilde{W}_{jk} \right)^{1/3} \]
\[ \tilde{W}_{ij} = \frac{W_{ij}}{\max\{W\}} \]

38. Zhang’s Local Clustering Coefficients
\[ \text{clusterZhang}_i = \frac{\sum_{j=1}^{N} \sum_{k=1}^{N} \tilde{W}_{ij}\tilde{W}_{ik}\tilde{W}_{jk}}{\left( \sum_{j=1}^{N} \tilde{W}_{ij} \right)^2 - \sum_{k=1}^{N} \tilde{W}_{ij}^2} \]

39. Holme’s Local Clustering Coefficients
\[ \text{clusterHolme}_i = \frac{\sum_{j=1}^{N} \sum_{k=1}^{N} \tilde{W}_{ij}\tilde{W}_{ik}\tilde{W}_{jk}}{\max\{W\} \cdot \sum_{j=1}^{N} \sum_{k=1}^{N} \tilde{W}_{ij}\tilde{W}_{jk}} \]

40. Edge-Weighted Interconnectivity
The edge-weighted interconnectivity is defined similarly with (#9) the unweighted interconnectivity. Firstly, the interconnectivity score for each edge is calculated.
\[ \text{EW\_ICN\_edge}_{ij} = \frac{2W_{ij} + \sum_{s \in \text{NN}(i) \cap \text{NN}(j)} W_{is}W_{sj}}{\sqrt{\text{strength}_i \cdot \text{strength}_j}} \]
\[ W_{ij} \] is the weight of the edge linking node \( i \) and node \( j \), and the previously defined \( \text{strength}_i \) is the sum of weights of connected edges to node \( i \).

Next, the edge-weighted interconnectivity for each node is calculated based on \( \text{EW\_ICN\_edge} \) scores.
\[ \text{EW\_ICN\_node}_i = \frac{1}{\text{deg}_i} \sum_{j=1}^{N} \text{EW\_ICN\_edge}_{ij} \]
Feature Category: Node-Weighted Properties

41. Node Weight

The node weight \( NW_i \) is directly extracted from the node weight matrix generated.

42. Node Weighted Cross Degree \(^{47}\)

For analyzing networks with heterogeneous node weights, the next two node-weighted informative measures were derived recently for the economic trading network study. In the definition, \( ExtA \) is the extended adjacency matrix, where \( ExtA_{ij} = A_{ij} + \delta_{ij} \), and \( \delta_{ij} \) is Kronecker’s delta constant.

\[
\delta_{ij} = \begin{cases} 
0, & \text{if } i \neq j \\
1, & \text{if } i = j 
\end{cases}
\]

\[
NW\text{crossdeg}_i = \sum_{j=1}^{N} ExtA_{ij} \cdot NW_i
\]

43. Node Weighted Local Clustering Coefficient \(^{47}\)

This node-weighted local clustering coefficient works, only if the node-weighted cross degree is not zero, otherwise the local clustering coefficient will be assumed as zero.

\[
NW\text{cluster}_i = \frac{1}{NW\text{crossdeg}_i} \sum_{j=1}^{N} \sum_{k=1}^{N} ExtA_{ij} \cdot NW_j \cdot ExtA_{jk} \cdot NW_k \cdot ExtA_{jk}
\]

44. Node-Weighted Neighbourhood Score \(^{15}\)

This score is defined in a study of disease-gene networks, by assigning the fold change of gene expression as the node weight. Below, \( \text{neighbour}(i) \) denotes all the neighbours of node \( i \) in the network.

\[
NW\text{neighbourhood}_i = \frac{1}{2} NW_i + \frac{1}{2} \cdot \frac{\sum_{j \in \text{neighbour}(i)} NW_j}{|\text{neighbour}(i)|}
\]

Feature Category: Directed Properties

45. In-Degree \(^{1,8}\)

As previously mentioned, “A” represents the undirected adjacency matrix and “a” represents the directed adjacency matrix, where \( a_{ij} = 1 \) means a directed edge has node \( i \) points to node \( j \). In-degree of a node counts the number of directed edges pointing to itself.

\[
\deg^+_i = \sum_{j \in N} a_{ji}
\]
46. **In-Degree Centrality**

The in-degree centrality for a node is the fraction of nodes its incoming edges are connected to.

47. **Out-Degree**

Out-degree of a node counts the number of directed edges pointing out of itself.

\[
deg_i^- = \sum_{j \in N} a_{ij}
\]

48. **Out-Degree Centrality**

The out-degree centrality for a node is the fraction of nodes its outgoing edges are connected to.

49. **Directed Local Clustering Coefficient**

In directed networks, local clustering coefficient is defined slightly different from undirected one.

\[
cluster_i^\pm = \frac{e_i}{(deg_i^+ + deg_i^-)(deg_i^+ + deg_i^- - 1)}
\]

50. **Neighbourhood Connectivity (only in)**

It is the average out-connectivity of all in-neighbours of node i.

\[
neighbourConnectivity_i^+ = \frac{\Sigma_{j \in N} a_{ij} \cdot deg_j^-}{\Sigma_{j \in N} a_{ij}}
\]

51. **Neighbourhood Connectivity (only out)**

It is the average in-connectivity of all out-neighbours of node i.

\[
neighbourConnectivity_i^- = \frac{\Sigma_{j \in N} a_{ij} \cdot deg_j^+}{\Sigma_{j \in N} a_{ij}}
\]

52. **Neighbourhood Connectivity (in & out)**

It is the average connectivity of all neighbours of node i, where the direction is ignored here.

\[
neighbourConnectivity_i^\pm = \frac{\Sigma_{j \in N} a_{ij} \cdot (deg_j^+ + deg_j^-) + \Sigma_{j \in N} a_{ji} \cdot (deg_j^+ + deg_j^-)}{\Sigma_{j \in N} a_{ij} + \Sigma_{j \in N} a_{ji}}
\]

53. **Average Directed Neighbour Degree**

\[
\text{avgDirectedNeighbourDeg}_i^\pm = \frac{\Sigma_{j \in N}[(a_{ij} + a_{ji}) \cdot (deg_j^+ + deg_j^-)]}{2 \cdot (deg_i^+ + deg_i^-)}
\]
D.2 Network-Level Descriptors

**Feature Category: Connectivity/Adjacency-based Properties**

1. **Number of Nodes**
   The number of the nodes (or vertices) in the network, noted as \( N \).

2. **Number of Edges**
   The number of edges (or links) in the network, noted as \( E \).

3. **Number of Selfloops**
   \[
   \text{selfloops}_G = \sum_{i=1}^{N} \text{selfloop}_i
   \]

4. **Maximum Connectivity**
   \[
   \text{connectivityMax}_G = \max\{\text{deg}_G\}
   \]

5. **Minimum Connectivity**
   \[
   \text{connectivityMin}_G = \min\{\text{deg}_G\}
   \]

6. **Average Number of Neighbors**
   The average of the number of neighbours (or degree, connectivity) for all nodes.
   \[
   \text{neighbourAvg}_G = \frac{1}{N} \sum_{i=1}^{N} \text{deg}_i
   \]

7. **Total Adjacency**
   The total adjacency is the half of the sum of the adjacency matrix entries.
   \[
   \text{totalAdjacency}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij}
   \]

8. **Network Density**
   The network density measures the efficiency of the information progression in a network in time. The denominator \( N*(N-1)/2 \) is the maximum number of links if the network is completely connected. For a directed network, the denominator is \( N*(N-1) \).
   \[
   \text{density}_G = \frac{E}{N(N-1)/2}
   \]

9. **Global Clustering Coefficient**
   Network clustering coefficient is the average of all the node-level clustering coefficients.
   \[
   \text{cluster}_G = \frac{1}{N} \sum_{i=1}^{N} \text{cluster}_i
   \]
10. Transitivity

Transitivity is calculated based on the number of triangles for each node in the network.

\[
\text{transitivity}_G = \frac{2 \cdot \sum_{i=1}^{N} \text{tri}_i}{\sum_{i=1}^{N} \text{deg}_i(\text{deg}_i - 1)}
\]

11. Heterogeneity

Heterogeneity measures the variation of degree distribution, reflecting the tendency of a network to have hubs. This index is biologically meaningful, as biological networks are usually heterogeneous with some central nodes highly connected and the rest nodes having few connections in the network.

\[
\text{heterogeneity}_G = \sqrt{\frac{N \cdot \sum_{i=1}^{N} (\text{deg}_i)^2}{(\sum_{i=1}^{N} \text{deg}_i)^2} - 1}
\]

12. Degree Centralization

Degree centralization (or connectivity centralization) is useful for distinguishing such characteristics as highly connected networks (e.g. star-shaped) or decentralized networks, which have been used for studying the structural differences of metabolic networks.

\[
\text{centralizationDeg}_G = \frac{N}{N - 2} \left( \frac{\text{connectivityMax}_G}{N - 1} - \text{density}_G \right)
\]

13. Central Point Dominance

Central point dominance is defined based on the measure of betweenness centrality.

\[
\text{centralDominance}_G = \frac{1}{N - 1} \sum_{i=1}^{N} (\max\{\text{centralityBtw}_i\} - \text{centralityBtw}_i)
\]

14. Degree Assortativity Coefficient

It measures the similarity of degree with respect to each edge in the network, by calculating the standard Pearson correlation coefficient between the degrees of the two connecting vertices of each edge. Its value lies in between -1 and 1, where 1 represents perfect assortativity and -1 indicates perfect dissortativity.

Feature Category: Shortest Path Length-based Properties

15. Total Distance

It is the sum of all the non-redundant pairwise shortest path distances in the network.

\[
\text{totalDistance}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij}
\]
16. **Network Diameter**

The network diameter is the largest distance in shortest path length matrix.

\[ \text{diameter}_G = \max\{D_{ij}\} \]

17. **Network Radius**

The network radius is the smallest distance in shortest path length matrix.

\[ \text{radius}_G = \min\{D_{ij}\} \]

18. **Shape Coefficient**

The shape coefficient of a network is defined by its radius and its diameter.

\[ \text{shapeCoef}_G = \frac{\text{diameter}_G - \text{radius}_G}{\text{radius}_G} \]

19. **Characterisitc Path Length**

The characteristic path length is the average distance in shortest path length matrix.

\[ \text{CPL}_G = \frac{\sum_{i=1}^{N} \text{avgSPL}_i}{N} \]

20. **Network Eccentricity**

\[ \text{eccentricity}_G = \sum_{i=1}^{N} \text{eccentricity}_i \]

21. **Average Eccentricity**

\[ \text{eccentricityAvg}_G = \frac{\text{eccentricity}_G}{N} \]

22. **Network Eccentric**

\[ \text{eccentric}_G = \frac{1}{N} \sum_{i=1}^{N} \text{eccentric}_i \]

23. **Eccentric Connectivity**

It is defined as the sum of the product of eccentricity and degree of each node, it has been shown the high correlation with regard to physical properties of diverse nature in various datasets.

\[ \text{eccentricConnect}_G = \sum_{i=1}^{N} \text{eccentric}_i \cdot \text{deg}_i \]

24. **Unipolarity**

It measures the minimal distance sum, which is the sum of shortest path lengths for each node.

\[ \text{unipolarity}_G = \min\{\text{distSum}_i\} \]

25. **Integration**

It is the sum of all the nodes’ distance sum, where each shortest path is counted once.
\[
\text{integration}_G = \frac{1}{2} \sum_{i=1}^{N} \text{distSum}_i
\]

26. Variation \(^{21}\)

The network variation is defined as the maximum variance in the node-level measures.

\[
\text{variation}_G = \max\{\text{deviation}_i\}
\]

27. Average Distance \(^{21}\)

This measures the mean shortest path length by dividing the integration by the number of nodes.

\[
\text{distAvg}_G = \frac{2 \cdot \text{integration}_G}{N}
\]

28. Mean Distance Deviation \(^{21}\)

This mean distance deviation is to average the node-level distance deviation values.

\[
\text{distDevMean}_G = \frac{1}{N} \sum_{i=1}^{N} \text{distDev}_i
\]

29. Centralization \(^{21}\)

This centralization descriptor sums the variance value for all nodes in the network.

\[
\text{centralization}_G = \sum_{i=1}^{N} \text{deviation}_i
\]

30. Global Efficiency \(^{54}\)

The global efficiency is a measure of the information exchange efficiency across the entire network. It can be used to determine the cost-effectiveness of the network structure.

\[
\text{efficiency}_G = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} \frac{1}{D_{ij}}
\]

**Feature Category: Topological Indices**

31. Edge Complexity Index \(^{48}\)

The global edge complexity is defined by dividing the total adjacency by \(N^2\).

\[
\text{edgeComplexity}_G = \frac{\text{totalAdjacency}_G}{N^2}
\]

32. Randic Connectivity Index \(^{55}\)

The randic index is a function of the connectivity of edges.

\[
\text{randic}_G = \sum_{e_{ij} \in E} \left( \text{deg}_i \cdot \text{deg}_j \right)^{-\frac{1}{2}}
\]
33. Atom-Bond Connectivity Index \(^{56}\)

The ABC index is a graph-invariant measure, which has been applied to study the stability of chemical structure. Here, it is used to describe the stability of a network structure.

\[
ABC_G = \sum_{e_{ij} \in G} \left( \frac{\deg_i + \deg_j - 2}{\deg_i \cdot \deg_j} \right)^2
\]

34. Zagreb Index \(^1\) \(^{57,58,59,60}\)

There are five Zagreb indices variants are defined based on the nodes’ degree.

\[
zagreb1_G = \sum_{i=1}^{N} \deg_i^2
\]

35. Zagreb Index 2

\[
zagreb2_G = \sum_{e_{ij} \in G} \deg_i \cdot \deg_j
\]

36. Modified Zagreb Index

\[
azgrebModified_G = \sum_{e_{ij} \in G} \frac{1}{\deg_i \cdot \deg_j}
\]

37. Augmented Zagreb Index

\[
zagrebAugmented_G = \sum_{e_{ij} \in G} \left( \frac{\deg_i \cdot \deg_j}{\deg_i + \deg_j - 2} \right)^3
\]

38. Variable Zagreb Index

\[
zagrebVariable_G = \sum_{e_{ij} \in G} \frac{\deg_i + \deg_j - 2}{\deg_i \cdot \deg_j}
\]

39. Narumi-Katayama Index \(^{61}\)

The NK index is the product of degrees of all nodes. It has been shown the relationships with thermodynamics properties. Additionally, its logged index, geometric index, and harmonic index are provided as follows. In our program, if Narumi index goes beyond \(\text{sys.maxsize}\), then Narumi Index and Narumi Geometric Index will be assigned as zero.

\[
narumi_G = \prod_{i=1}^{N} \deg_i
\]

40. Narumi-Katayama Index (log)

\[
narumiLog_G = \log_2 \left( \prod_{i=1}^{N} \deg_i \right)
\]
41. Narumi Geometric Index

\[ \text{narumiGe}_G = \left( \prod_{i=1}^{N} \text{deg}_i \right)^{\frac{1}{N}} \]

42. Narumi Harmonic Index

\[ \text{narumiHar}_G = \frac{N}{\sum_{i=1}^{N} (\text{deg}_i)^{-1}} \]

43. Alpha Index

Alpha index is a connectivity measure to evaluate the number of cycles in a network in comparison with maximum number of cycles, such that the higher alpha index, the more connected nodes. Trees and simple networks have alpha index equal to zero, and a completely connected network have alpha index equal to 1. Sometimes, alpha index is named as Meshedness Coefficient.

\[ \alpha_G = \frac{E - N}{\frac{N(N-1)}{2} - (N-1)} \]

44. Beta Index

It measures the graph connectivity, by the ratio of the number of edges over the number of nodes. Simple networks have beta value less than 1, and more complex networks have higher beta index.

\[ \beta_G = \frac{E}{N} \]

45. Pi Index

Pi is the relationship between the total length of the network and its diameter. Namely Pi index, it has a similar meaning with the definition of \( \pi \), indicating of the shape of the network.

\[ \pi_G = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij}}{\text{diameter}_G} \]

46. Eta Index

Eta index is the average adjacency per edge. Adding nodes will result in decreasing of eta index.

\[ \eta_G = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij}}{E} \]

47. Hierarchy

Hierarchy index is the gradient of the linear power-law regression, by fitting \( \log_{10}(\text{node frequency}) \) over \( \log_{10}(\text{degree distribution}) \). It usually has the value between 1 and 2, where the low hierarchy indicates the weak hierarchical relationship.

Hierarchy is notated as \( h \) in the fitted regression equation \( y=ax^h \), where \( x \) is the degree distribution and \( y \) is the node frequency of that specific degree.

\[ y = a \cdot x^{\text{hierarchy}} \]
48. Robustness

Robustness is to measure the stability of a network under node-removal attacks. By removing each node, the size of the largest fragmented component $S$ is used to define the robustness.

$$\text{robustness}_G = \frac{\sum_{k=1}^N S_k}{N(N-1)}$$

49. Medium Articulation

Medium articulation MA is a complexity measure of a network, reaching its maximum with medium number of edges. It is defined based on the redundancy (MA$_R$) and the mutual information (MA$_I$).

$$MA_G = MA_R \cdot MA_I$$

Redundancy MA$_R$ is defined as:

$$MA_R = 4 \left( \frac{R - R_{\text{path}}}{R_{\text{clique}} - R_{\text{path}}} \right) \left( 1 - \frac{R - R_{\text{path}}}{R_{\text{clique}} - R_{\text{path}}} \right)$$

$$R = \frac{1}{E} \sum_{i=1}^{N} \sum_{j \neq i}^{N} \log_{10} (\text{deg}_i \cdot \text{deg}_j)$$

$$R_{\text{clique}} = 2 \cdot \log_{10} (N - 1)$$

$$R_{\text{path}} = 2 \cdot \frac{N - 2}{N - 1} \log_{10} 2$$

Mutual information MA$_I$ is defined as:

$$MA_I = 4 \left( \frac{I - I_{\text{clique}}}{I_{\text{path}} - I_{\text{clique}}} \right) \left( 1 - \frac{I - I_{\text{clique}}}{I_{\text{path}} - I_{\text{clique}}} \right)$$

$$I = \frac{1}{E} \sum_{i=1}^{N} \sum_{j \neq i}^{N} \log_{10} \left( \frac{2E}{\text{deg}_i \cdot \text{deg}_j} \right)$$

$$I_{\text{clique}} = \log_{10} \left( \frac{N}{N-1} \right)$$

$$I_{\text{path}} = \log_{10} (N - 1) - \frac{N - 3}{N - 1} \log_{10} 2$$

50. Complexity Index A

It is the ratio of total adjacency and the total distance of a network.

$$\text{complexity}_{A_G} = \frac{\text{totalAdjacency}_G}{\text{totalDistance}_G}$$

51. Complexity Index B

It is defined by the ratio of vertex degree and its distance sum for each vertex.

$$\text{complexity}_{B_G} = \sum_{i=1}^{N} \frac{\text{deg}_i}{\text{distSum}_i}$$

52. Wiener Index

The Wiener index measures the sum of the shortest path lengths between all pairs of vertices.
$$\text{wiener}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij}$$

53. Hyper-Wiener Index 68

$$\text{hyperWiener}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (D_{ij}^2 + D_{ij})$$

54. Harary Index 1 69

$$\text{harary1}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij}^{-1}$$

55. Harary Index 2 69

$$\text{harary2}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij}^{-2}$$

56. Compactness 70

This measure is based on Wiener index, by dividing the Wiener index by \(N(N-1)\).

$$\text{compactness}_G = \frac{4 \cdot \text{wiener}_G}{N(N - 1)}$$

57. Superpendentic Index 71

$$\text{superpendentic}_G = \left( \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij} \right)^{\frac{1}{2}}$$

58. Hyper-Distance-Path Index 72,73

This index is consist of two parts: the exactly Wiener index, and the delta number.

$$\text{hyper} \_ \text{path}_G = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} D_{ij} + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left( \frac{D_{ij}}{2} \right)$$

59. BalabanJ Index 74

This BalabanJ index counts into the distance sum of the two end-vertex for each edge. BalabanJ index has been proven to be relevant to the network branching. There are another three differently defined variants of BalabanJ indices are given in the followings.

$$Jm_G = \frac{E}{\mu + 1} \sum_{e_i \in G} (\text{disSum}_i \cdot \text{disSum}_i)^{-\frac{1}{2}}$$

Where, \(\mu = E + 1 - N\), which denotes the cyclomatic number of a graph.

60. BalabanJ-Like Index 1 75

$$Jm1_G = \frac{E}{\mu + 1} \sum_{e_i \in G} (\text{disSum}_i \cdot \text{disSum}_i)^{\frac{1}{2}}$$
61. BalabanJ-Like Index 2
\[ Jm2_G = \frac{E}{\mu + 1} \sum_{E_{i,j} \in G} \left( \frac{\text{disSum}_i + \text{disSum}_j}{\sqrt{\text{disSum}_i \cdot \text{disSum}_j}} \right)^{1/2} \]

62. BalabanJ-Like Index 3
\[ Jm3_G = \frac{E}{\mu + 1} \sum_{E_{i,j} \in G} \left( \frac{\text{disSum}_i \cdot \text{disSum}_j}{\text{disSum}_i + \text{disSum}_j} \right)^{1/2} \]

63. Geometric Arithmetic Index 1
GA index consists of the geometrical and arithmetic means of the end-to-end degree of an edge.
\[ GA1_G = \sum_{E_{i,j} \in G} 2\sqrt{\text{deg}_i \cdot \text{deg}_j} \]

64. Geometric Arithmetic Index 2
There are 2 extended geometric-arithmetic indices, which make use of the information of the shortest path lengths. In some studies, the geometric-arithmetic indices have shown its power in characterizing the network structure features.
\[ GA2_G = \sum_{E_{i,j} \in G} 2\sqrt{\frac{n_i \cdot n_j}{n_i + n_j}} \]
\[ n_i := |x \in \text{node}(G), D_{xi} < D_{xj}| \]
\[ n_j := |x \in \text{node}(G), D_{xj} < D_{xi}| \]
In the definition of geometric arithmetic index 2 (GA2), x is a node, \( n_i \) is the number of nodes closer to node \( i \), and \( n_j \) is the number of nodes closer to node \( j \), while the nodes with same distance to node \( i \) and node \( j \) are ignored.

65. Geometric Arithmetic Index 3
\[ GA3_G = \sum_{E_{i,j} \in G} 2\sqrt{\frac{m_i \cdot m_j}{m_i + m_j}} \]
\[ m_i := |y \in \text{edge}(G), D_{yi} < D_{yj}| \]
\[ m_j := |y \in \text{edge}(G), D_{yj} < D_{yi}| \]
In the definition of geometric arithmetic index 3 (GA3), y is an edge in the graph, the distance between edge \( y \) to node \( i \) is defined as \( D_{yi} = \min \{D_{pi}, D_{qi}\} \), where \( p \) and \( q \) are the two ends of edge \( y \). In the context above, \( m_i \) is number of edges closer to node \( i \) and \( m_j \) is the number of edges closer to node \( j \), while the edges with same distance to node \( i \) and node \( j \) are not counted.

66. Szeged Index
\[ szeugged_G = \sum_{E_{i,j} \in G} n_i \cdot n_j \]
Where \( n_i \) and \( n_j \) are as same defined as the previous geometric-arithmetic index 2.
67. Product of Row Sums

If PRS is greater than `sys.maxsize`, it will be assigned as zero in the program.

\[
PRS_G = \prod_{i=1}^{N} \text{distSum}_i
\]

68. Product of Row Sums (log)

\[
PRS\log_G = \log_2 \left( \prod_{i=1}^{N} \text{distSum}_i \right)
\]

69. Schultz Topological Index

By using adjacency matrix \( A \), shortest path distance matrix \( D \), and the vertex degree vector \( v \), Schultz defined a topological index to describe the network structure.

In the equation below, \((D+A)\) forms an additive \( N \times N \) matrix, and this matrix is then multiplied by a \( 1 \times N \) vector \( v \), such that obtaining another \( 1 \times N \) vector. The sum of all the elements in the resultant vector is called the Schultz topological index.

\[
schultz_G = \sum_{i=1}^{N} [v(D+A)]_i
\]

70. Gutman Topological Index

Gutman topological index is a further defined Schultz index, where \( ADA \) is the matrix multiplication.

\[
gutman_G = \sum_{i=1}^{N} \sum_{j=1}^{N} [ADA]_{ij}
\]

71. Efficiency Complexity

The efficiency complexity is motivated in analyzing the weighted networks, as it suggests to measure not only the shortest path lengths but also the cost (number of links).

\[
EC_G = 4 \left( \frac{E - E_{path}}{1 - E_{path}} \right) \left( 1 - \frac{E - E_{path}}{1 - E_{path}} \right)
\]

\[
E = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} \frac{1}{D(i,j)}
\]

\[
E_{path} = \frac{2}{N(N-1)} \sum_{i=1}^{N} \left( N - \frac{N - i}{i} \right)
\]
**Feature Category: Entropy-Based Complexity Indices**

72. **Information Content (Degree Equality)**

This information content measures the probability distribution of vertex degree, where \( N_i \) is the number of nodes having the same degree, and \( k^d \) is the maximum of degree.

\[
I_{\text{vertexDegree}} = - \sum_{i=1}^{k^d} \frac{N_i}{N} \cdot \log_2 \left( \frac{N_i}{N} \right)
\]

73. **Information Content (Edge Equality)**

This measure is based on the probability distribution of edge connectivity, where each edge has an end-to-end connectivity value. Let \((a, b)\) and \(a \leq b\) be the edge’s end-to-end connectivity, such that the edges having the same edge connectivity will be grouped into the same subset.

\[
I_{\text{edgeEquality}} = - \sum_{i=1}^{k^{\text{edge}}} \frac{E_i}{E} \cdot \log_2 \left( \frac{E_i}{E} \right)
\]

Where, \( E_i \) is the number of edges having the same end-to-end connectivity, and \( k^{\text{edge}} \) is the number of different edge subsets.

74. **Information Content (Edge Magnitude)**

As another measure based on the edge information, it is defined by the connectivity magnitude of each edge, and \( \text{randic}_G \) is the network-level randic connectivity index introduced previously.

\[
I_{\text{edgeMagnitude}} = - \sum_{(i,j) \in G} \frac{(\text{deg}_i \cdot \text{deg}_j)^{1/2}}{\text{randic}_G} \cdot \log_2 \left( \frac{(\text{deg}_i \cdot \text{deg}_j)^{1/2}}{\text{randic}_G} \right)
\]

75. **Information Content (Distance Degree)**

The distance degree of a node \( i \) is equivalently the distance sum \( \text{distSum}_i \) defined previously.

\[
I_{\text{distanceDegree}} = - \sum_{i=1}^{N} \frac{\text{distSum}_i}{2 \cdot \text{Weiner}_G} \cdot \log_2 \left( \frac{\text{distSum}_i}{2 \cdot \text{Weiner}_G} \right)
\]

76. **Information Content (Distance Degree Equality)**

The probability distribution regarding on the nodes’ distance degree value gives the definition of the information content on distance degree equality. As below, \( k^{dd} \) is the number of node groups in the distribution of distance degree, \( N_i^{dd} \) is the number of nodes having the same distance degree.

\[
I_{\text{distanceDegreeEquality}} = - \sum_{i=1}^{k^{dd}} \frac{N_i^{dd}}{N} \cdot \log_2 \left( \frac{N_i^{dd}}{N} \right)
\]

77. **Radial Centric Information Index**

It is measuring the probability distribution of vertex eccentricity. In the definition below, \( N^e_i \) is the number of nodes having the equal eccentricity value \( i \), and \( k^e \) is the maximum of eccentricity.

\[
I_{\text{radialCentric}} = - \sum_{i=1}^{k^e} \frac{N^e_i}{N} \cdot \log_2 \left( \frac{N^e_i}{N} \right)
\]
78. Distance Degree Compactness

This measure is defined based on the distribution of nodes’ locations from the centre of a network, where the centre is determined by the closeness centrality score in this case. Here, \( Q_k \) is the sum of distance degree of all nodes that located at the same topological distance \( k \) from the centre.

\[
I_{\text{compactness}} = 2\text{Weiner}_G \cdot \log_2(2\text{Weiner}_G) - \sum_k Q_k \cdot \log_2(Q_k)
\]

79. Distance Degree Centric Index

\[
I_{\text{distanceDegreeCentric}} = -\sum N_i \cdot \log_2(N_i) - K_i \cdot \log_2(K_i)
\]

Where \( N_i \) is the number of nodes having the same eccentricity and the same degree, \( K_i \) is the number of equivalent classes of \( N_i \).

80. Graph Distance Complexity

As a similar definition as \( I_{\text{infoLayer}} \), this distance complexity includes the nodes’ distance sums.

\[
I_{\text{distanceComplexity}} = -\sum \frac{1}{N} \cdot \log_2\left(\frac{1}{N}\right) + \sum \frac{2k_i}{N^2} \cdot \log_2\left(\frac{2k_i}{N^2}\right)
\]

81. Information Layer Index

\[
I_{\text{infoLayer}} = -\sum \frac{1}{N} \cdot \log_2\left(\frac{1}{N}\right) - \sum \frac{2k_i}{N^2} \cdot \log_2\left(\frac{2k_i}{N^2}\right)
\]

Where, \( ecc_i \) is the eccentricity value of node \( i \), and \( N_i^j \) is the number of nodes in the \( j^{th} \) sphere of node \( i \). In other words, \( N_i^j \) is the number of nodes in shorest distance \( j \) away from node \( i \).

82. Bochev Information Index 1

Bochev indices applies the probability distribution of shortest path lengths to Shannon’s entropy formula, and it has three variants as follows. \( diameter_G \) is the maximum distance between two nodes in the network, and \( k_i \) is the occurrence of distance \( i \) in the shortest path length matrix \( D_{ij} \).

\[
I_{\text{bochev1}} = -\frac{1}{N} \cdot \log_2\left(\frac{1}{N}\right) - \sum \frac{2k_i}{N^2} \cdot \log_2\left(\frac{2k_i}{N^2}\right)
\]

83. Bochev Information Index 2

\[
I_{\text{bochev2}} = -\text{Weiner}_G \cdot \log_2(\text{Weiner}_G) - \sum i \cdot k_i \cdot \log_2(i)
\]

84. Bochev Information Index 3

\[
I_{\text{bochev3}} = -\sum \frac{2k_i}{N(N-1)} \cdot \log_2\left(\frac{2k_i}{N(N-1)}\right)
\]
85. Balaban-like Information Index 1 \(^{89,90}\)

Differently from BalabanJ indices, Balaban-like information index 1 & 2 are defined based on the distribution of distance degree in the network. In the equation below, \(g_k\) is the number of nodes at distance \(k\) from node \(i\), and \(\mu\) is namely the cyclomatic number.

\[
I_{balaban1} = \frac{E}{\mu + 1} \sum_{\{i,j\} \in G} [u_i \cdot u_j]^{-1/2}
\]

\[
u_i = -\sum_{k=1}^{\text{diameter}} \frac{k \cdot g_k}{\text{distSum}_k} \cdot \log_2\left(\frac{k}{\text{distSum}_k}\right)
\]

\[
\mu = E + 1 - N
\]

86. Balaban-like Information Index 2 \(^{89,90}\)

\[
I_{balaban2} = \frac{E}{\mu + 1} \sum_{\{i,j\} \in G} [v_i \cdot v_j]^{-1/2}
\]

\[
v_i = \text{distSum}_i \cdot \log_2(\text{distSum}_i) - u_i
\]

---

**Feature Category: Eigenvalue-Based Complexity Indices**

87. Graph Energy \(^{91}\)

Given a network, let \(\{\lambda_1, \lambda_2, ... \lambda_k\}\) be the non-zero eigenvalues of its adjacency matrix, such that \(k\) is the number of eigenvalues and \(\lambda_{\text{max}}\) is the maximum of the eigenvalues.

\[
\text{Energy}_G = \sum_{i=1}^{k} |\lambda_i|
\]

88. Laplacian Energy \(^{91}\)

Laplacian matrix \(L\) is generated based on the degree and the adjacency relationships, as below. Such that, Laplacian matrix produces \(\mu_i : \{\mu_1, \mu_2, ..., \mu_k\}\) as the Laplacian eigenvalues of the network.

\[
L_{ij} = \begin{cases} -1 & \text{if } A_{ij} = 1 \\ \text{deg}_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}
\]

\[
\text{LaplacianEnergy}_G = \sum_{i=1}^{k} |\mu_i - \frac{2E}{N}|
\]

89. Spectral Radius \(^{92}\)

\[
\text{SpRadius}_G = \max(|\lambda_i|)
\]

90. Estrada Index \(^{93}\)

\[
\text{Estrada}_G = \sum_{i=1}^{k} e^{\lambda_i}
\]
91. **Laplacian Estrada Index** 94

\[ \text{Laplacian Estrada}_G = \sum_{i=1}^{k} e^{\mu_i} \]

92. **Quasi-Wiener Index** 95

Quasi-Wiener is defined by Laplacian eigenvalues. As the last eigenvalue \( \mu_k \) is always zero, it is excluded.

\[ \text{quasiWiener}_G = N \sum_{i=1}^{k-1} \frac{1}{\mu_i} \]

93. **Mohar Index 1** 73,96

\[ \text{mohar1}_G = \frac{1}{N} \cdot \text{quasiWiener}_G \cdot \log_2 \left( \sum_{i=1}^{k-1} \mu_i \right) \]

94. **Mohar Index 2** 73,96

\[ \text{mohar2}_G = \frac{4}{N \cdot \mu_{k-1}} \]

95. **Graph Index Complexity** 64

\[ C_{r_G} = 4 \cdot c_r \cdot (1 - c_r) \]

\[ c_r = \frac{\lambda_{\text{max}} - 2 \cos \frac{\pi}{N+1}}{N - 1 - 2 \cos \frac{\pi}{N+1}} \]

96 - 195. **A Set of Eigenvalue-Based Descriptors from Variants of Matrices** 97,98

There are 5 novel eigenvalue-based descriptors recently introduced, namely \( H_{MG}, S_{MG}, IS_{MG}, PM_{MG}, \) and \( IP_{MG} \). Let \( M \) be a re-defined matrix based on the given graph \( G \), and \( \{ \lambda_1, \lambda_2, \ldots, \lambda_k \} \) be its non-zero eigenvalues. As the factor \( "s" \) may have different discrimination power for different networks, we thus provide these eigenvalue-based descriptors at both \( s = 1 \) and \( s = 2 \).

\[ H_{MG} = -\sum_{i=1}^{k} \left[ \frac{|\lambda_i|^2}{\sum_{j=1}^{k} |\lambda_j|^2} \log_2 \left( \frac{|\lambda_i|^2}{\sum_{j=1}^{k} |\lambda_j|^2} \right) \right] \]

\[ S_{MG} = \sum_{i=1}^{k} |\lambda_i|^{-\frac{1}{2}} \]

\[ IS_{MG} = \frac{1}{\sum_{i=1}^{k} |\lambda_i|^{-\frac{1}{2}}} \]

\[ PM_{MG} = \prod_{i=1}^{k} |\lambda_i|^{-\frac{1}{2}} \]

\[ IP_{MG} = \frac{1}{\prod_{i=1}^{k} |\lambda_i|^{-\frac{1}{2}}} \]

These 5 eigenvalue-based descriptors could be applied to the following 10 differently re-defined matrices, including (1) adjacency matrix, (2) laplacian matrix, (3) distance matrix, (4) distance
path matrix, (5) augmented vertex degree matrix, (6) extended adjacency matrix, (7) vertex connectivity matrix, (8) random walk Markov matrix, (9) weighted structure function matrix 1, and (10) weighted structure function matrix 2, which are defined as follows.

Therefore, totally 50 eigenvalue-based descriptors are calculated in this set.

(1) Adjacency matrix $A_{ij}$ is the initially generated based on the connections of the network.
(2) Laplacian matrix $L_{ij}$ is introduced previously in the definition of Laplacian energy.
(3) Distance matrix $D_{ij}$ is the shortest distance between all the nodes.
(4) Distance path matrix $DP_{ij}$, is derived from the distance matrix, by counting all the internal paths between a pair of nodes, including their shortest paths.

$$DP_{ij} = \left( D_{ij} + 1 \right)$$

(5) Augmented vertex degree matrix $AVD_{ij}$ is defined by the nodes’ degree and distance.

$$AVD_{ij} = \frac{\text{deg}_j}{2^{\text{deg}_j}}$$

(6) Extended adjacency matrix $EA_{ij}$ is a symmetric matrix based on the nodes’ degree.

$$EA_{ij} = \begin{cases} \frac{1}{2} \left( \frac{\text{deg}_i}{\text{deg}_j} + \frac{\text{deg}_j}{\text{deg}_i} \right) & \text{if } A_{ij} = 1 \\ 0 & \text{otherwise} \end{cases}$$

(7) Vertex connectivity matrix $VC_{ij}$, is another symmetric matrix based on the nodes’ degree.

$$VC_{ij} = \begin{cases} 1 & \text{if } A_{ij} = 1 \\ \frac{1}{\text{deg}_i \cdot \text{deg}_j} & \text{otherwise} \end{cases}$$

(8) Radom walk Markov matrix $RWM_{ij}$ is a non-symmetric matrix based on the nodes’ degree. It is based on the assumption that each neighbour node can be reached from a given node with the same probability, such that the probability of reaching the neighbor of node $i$ is $1/\text{deg}_i$. The generated distribution of walks is called the simple random walks.

$$RWM_{ij} = \begin{cases} 1 & \text{if } A_{ij} = 1 \\ \frac{1}{\text{deg}_i} & \text{otherwise} \end{cases}$$

(9) Weighted structure function matrix 1 $IM1_{ij}$, is a more complexly defined matrix. In the following definitions, $\text{radius}_G$ is the maximum shortest path length in the network, and $|S_d(i)|$ is the number of nodes that are at the shortest distance $d$ away from the node $i$.

$$f1(i) = \sum_{d=1}^{\text{radius}_G} (\text{radius}_G + 1 - d) \cdot |S_d(i)|$$

$$pf1(i) = \frac{f1(i)}{\sum_j f1(j)}$$

$$IM1_{ij} = 1 - \frac{|pf1(i) - pf1(j)|}{2^{\text{deg}_ij}}$$
(10) Weighted structure function matrix 2 $IM_{2(i)}$ is slight differently defined as below.

\[
    f_2(i) = \sum_{d=1}^{\text{radius}_G} (\text{radius}_G \cdot e^{1-d}) \cdot |S_d(i)|
\]

\[
    pf_2(i) = \frac{f_2(i)}{\sum_{j=1}^{N} f_2(j)}
\]

\[
    IM_{2 ij} = 1 - \frac{|pf_2(i) - pf_2(j)|}{2^{\delta_{ij}}}
\]

**Feature Category: Edge-Weighted Properties**

196. Weighted Transitivity \(^8\)

\[
    \text{weighted_transitivity}_G = \frac{\sum_{i=1}^{N} \text{geo}_\text{tri}_i}{\sum_{i=1}^{N} \text{deg}_i(\text{deg}_i - 1)}
\]

197. Barrat’s Global Clustering Coefficients \(^43\)

\[
    \text{clusterBarrat}_G = \frac{1}{N} \sum_{i=1}^{N} \text{clusterBarrat}_i
\]

198. Onnela’s Global Clustering Coefficients \(^{43,44}\)

\[
    \text{clusterOnnela}_G = \frac{1}{N} \sum_{i=1}^{N} \text{clusterOnnela}_i
\]

199. Zhang’s Global Clustering Coefficients \(^{43,45}\)

\[
    \text{clusterZhang}_G = \frac{1}{N} \sum_{i=1}^{N} \text{clusterZhang}_i
\]

200. Holme’s Global Clustering Coefficients \(^{43,46}\)

\[
    \text{clusterHolme}_G = \frac{1}{N} \sum_{i=1}^{N} \text{clusterHolme}_i
\]

**Feature Category: Node-Weighted Properties**

201. Total Node Weight

\[
    \text{total}_\text{NW}_G = \sum_{i=1}^{N} \text{NW}_i
\]

202. Node Weighted Global Clustering Coefficient \(^{47}\)

\[
    \text{NWcluster}_G = \frac{1}{N} \sum_{i=1}^{N} \text{NWcluster}_i
\]
Feature Category: Directed Properties

203. Average In-Degree

\[
\text{avg}_{\text{deg}}^+ = \frac{1}{N} \sum_{i \in N} \text{deg}_i^+
\]

204. Maximum In-Degree

\[
\text{max}_{\text{deg}}^+ = \max\{\text{deg}_i^+\}
\]

205. Minimum In-Degree

\[
\text{min}_{\text{deg}}^+ = \min\{\text{deg}_i^+\}
\]

206. Average Out-Degree

\[
\text{avg}_{\text{deg}}^- = \frac{1}{N} \sum_{i \in N} \text{deg}_i^-
\]

207. Maximum Out-Degree

\[
\text{max}_{\text{deg}}^- = \max\{\text{deg}_i^-\}
\]

208. Minimum Out-Degree

\[
\text{min}_{\text{deg}}^- = \min\{\text{deg}_i^-\}
\]

209. Directed Global Clustering Coefficient \(^1\)

\[
\text{cluster}_G^\pm = \frac{1}{N} \sum_{i \in N} \text{cluster}_i^\pm
\]

230. Directed Flow Hierarchy \(^99\)

Flow hierarchy is a measurement of the percentage of edges that not involved in any directed cycles in the directed network.
D.3 Edge-Level Descriptors

1. **Edge Weight**

The edge weight $EW_i$ is directly extracted from the user-provided edge weight list.

2. **Edge Betweenness** \(^{22,100}\)

Similarly with the definition of the node-level betweenness centrality. The edge betweenness quantifies the number of times an edge serving as a linking bridge along the shortest path between two nodes. In the following equation, node $s$ and node $t$ are two different nodes in the network, $\sigma_{st}(e)$ is the number of shortest paths from $s$ to $t$ that passing through the edge $e$, and $\sigma_{st}$ is the number of shortest paths from node $s$ to node $t$.

\[
\text{edgeBetweenness}_e = \frac{\sum_{st} \sigma_{st}(e)}{\sigma_{st}}
\]
(E) Computational Time Cost

The CPU time of PROFEAT was evaluated in computing the slim set of network descriptors for 10 human tissue-specific protein-protein-interaction (PPI) networks in five different network types and various sizes (Table 11).

These networks were constructed as follows: firstly 38,131 human PPIs between 9,084 proteins were collected from HPRD (Human Protein Reference Database)\(^\text{101}\), secondly 111,152 tissue-protein associations were extracted from HPRD, thirdly the human PPIs were grouped according to their distributed tissues, and lastly the tissue-specific lists of PPIs were processed to find their largest connected components as the human tissue-specific PPI networks. 10 tissue-specific PPI networks were selected with varying number of nodes from 63 to 2317 and varying number of edges from 91 to 4924. Each of the 10 networks was constructed into five different network types. The first four types are undirected unweighted, edge-weighted, node-weighted, and edge-node-weighted networks with their edge-weights or node-weights randomly generated. The fifth type is the directed unweighted network with the direction of each edge tentatively assigned from the left-node to the right-node in the input of SIF Format.

The evaluation of CPU time on running the 10 tested networks was measured on a Dell OptiPlex9010 Intel Core i7-3770 3.4GHz CPU, were summarized in Table 11 and Figure 2. Specifically, the CPU time for the un-weighted network is within 1 minute for a network having no more than 1500 nodes or 3000 edges, the CPU time is less than 3 minutes if the network size is less than 2400 nodes or 5000 edges. On the other hand, the CPU time for the edge-weighted network is less than 30 minutes if the network size is no more than 1,200 nodes or 2,200 edges. The edge-node-weighted network requires the highest computational time, as it costs about 1 hour for the network with 1,600 nodes or 3,300 edges. As there are only 23 descriptors calculated for directed networks, its CPU time is always within 5 seconds for all the testing networks.
Table 11 | CPU time in computing the slim set of PROFEAT network descriptors for ten human tissue-specific PPI networks in five different network types

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Human Systems</th>
<th>Network Size</th>
<th>CPU Time (Mins) for Different Network Types</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>No of Nodes</td>
<td>No of Edges</td>
</tr>
<tr>
<td>Lymph Node</td>
<td>Immune</td>
<td>63</td>
<td>91</td>
</tr>
<tr>
<td>Hippocampus</td>
<td>Nervous</td>
<td>107</td>
<td>146</td>
</tr>
<tr>
<td>Bone Marrow</td>
<td>Immune</td>
<td>189</td>
<td>348</td>
</tr>
<tr>
<td>Muscle</td>
<td>Musculoskeletal</td>
<td>315</td>
<td>632</td>
</tr>
<tr>
<td>Small Intestine</td>
<td>Digestive</td>
<td>616</td>
<td>980</td>
</tr>
<tr>
<td>Colon</td>
<td>Digestive</td>
<td>988</td>
<td>1951</td>
</tr>
<tr>
<td>Ovary</td>
<td>Reproductive</td>
<td>1165</td>
<td>2230</td>
</tr>
<tr>
<td>Spleen</td>
<td>Immune</td>
<td>1292</td>
<td>2543</td>
</tr>
<tr>
<td>Pancreas</td>
<td>Endocrine</td>
<td>1625</td>
<td>3336</td>
</tr>
<tr>
<td>Lung</td>
<td>Respiratory</td>
<td>2317</td>
<td>4942</td>
</tr>
</tbody>
</table>

Figure 2 | CPU time in computing the slim set of PROFEAT network descriptors for the networks described in Table 11 with respect to the number of nodes (left) and the number of edges (right)
### Typical Applications of Network Descriptors in Systems Biology

**Table 12** | List of network descriptors (node-level, network-level, and edge-level) in different categories provided by PROFEAT and the selected systems biological applications

<table>
<thead>
<tr>
<th>Network Descriptors</th>
<th>Applications in Systems Biology</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connectivity/Adjacency-based Properties</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Node-Level:</strong> Degree, Scaled Connectivity, Number of Selfloops/Triangles, Zscore, Clustering Coefficient, Topological Coefficient, Neighborhood Connectivity, Interconnectivity, Degree Centrality, Bridging Coefficient</td>
<td>Degree, average neighbours and density implicated the genes in disease network(^{102}). Neighbourhood connectivity measured the stability of protein/genetic regulatory networks(^{103}). Interconnectivity prioritized the disease genes(^{15}). Global clustering coefficient provided molecular characterization in gene co-expression network(^{104}).</td>
</tr>
<tr>
<td><strong>Network-Level:</strong> Number of Nodes/Edges/Selfloops, Max/Min Connectivity, Average Neighbours, Total Adjacency, Density, Average Clustering Coefficient, Transitivity, Heterogeneity, Degree Centralization, Central Point Dominance, Degree Assortativity Coefficient</td>
<td></td>
</tr>
<tr>
<td><strong>Edge-Level:</strong> Unweighted Edge Betweenness</td>
<td></td>
</tr>
<tr>
<td><strong>Shortest Path Length-based Properties</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Node-Level:</strong> Average Shortest Path Length, Eccentric, Eccentricity, Radiality, Distance Sum, Deviation, Distance Deviation, Closeness Centrality, Eccentricity Centrality, Harmonic Centrality, Residual Centrality, Load Centrality, Betweenness Centrality, Bridging Centrality, CurrentFlow Closeness, CurrentFlow Betweenness</td>
<td>Centrality and peripherality (eccentricity, radiality) implicated genes in disease network(^{102}). Eccentricity and distance deviation identified the metabolic biomarkers(^{105}). Shortest path length, betweenness, closeness, radiality and integration explored protein-drug interactome for lung cancer(^{106}), identified the hubs and bridging nodes in drug addiction mechanisms(^{107}). Edge-betweenness facilitated the modularity analysis(^{28}).</td>
</tr>
<tr>
<td><strong>Network-Level:</strong> Total Distance, Shape Coefficient, Diameter, Radius, Character. Path Length, Network Eccentricity, Average Eccentricity, Network Eccentric, Eccentric Connectivity, Unipolarity, Integration, Variation, Avg Distance, Mean Distance Deviation, Centralization, Global Efficiency</td>
<td></td>
</tr>
<tr>
<td><strong>Edge-Level:</strong> Edge Weight, Weighted Edge Betweenness</td>
<td></td>
</tr>
<tr>
<td><strong>Topological Indices</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Node-Level:</strong> N.A.</td>
<td>Exponent of power-law degree distribution (hierarchy index), provided molecular characterization of cellular state in gene co-expression network(^{104}), characterized the yeast genetic interaction network(^{108}), measured the robustness of protein interaction networks and genetic regulatory networks(^{103}). Complexity indices and BalabanJ index classified the metabolic networks from 3 domains of life(^{109}).</td>
</tr>
<tr>
<td><strong>Network-Level:</strong> Edge Complexity Index, Randic Connectivity Index, ABC Index, Zagreb Indices, Narumi Indices, Alpha/Beta/Pi/Eta Index, Hierarchy, Robustness, Medium Articulation, Complexity Indices, Wiener Index, Hyper-Wiener, Harary Indices, Compactness, Superpendentide Index, Hyper-Distance-Path Index, BalabanJ, BalabanJ-like Indices, Geometric Arithmetic Indices, Product of Row Sums, Topological Indices, Szeged Index, Efficiency Complexity</td>
<td></td>
</tr>
</tbody>
</table>
### Entropy-based Complexity Indices

| Node-Level: N.A. | Information-theoretic entropy measures identified and ranked the highly discriminating metabolic biomarker candidates for obesity\(^\text{105}\). Radial centric information index, degree equality-information index, and 3 Dehmer’s entropy descriptors classified the metabolic networks of 43 organisms from 3 domains of life\(^\text{109}\). |
| Network-Level: Entropy on (degree equality/edge equality/edge magnitude/distance degree/distance degree equality), Radial Centric Information Index, Distance Degree Compactness, Distance Degree Centric Index, Graph Distance Complexity, Info Layer Index, Bonchev Info Indices, Balaban-like Info Indices |

### Eigenvalue-based Complexity Indices

| Node-Level: Eigenvector Centrality, Page Rank Centrality | PageRank centrality identified prognostic marker genes for pancreatic cancer\(^\text{110}\). The PageRank centrality/degree quotient scored and found the non-hub important nodes in microbial networks from 3 distinct organisms\(^\text{37}\). |
| Network-Level: Spectral Radius, Estrada Index, Laplacian Estrada Index, Quasi-Weiner Index, Mohar Indices, Graph Index Complexity, 50 Dehmer’s Entropy by Matrices of (adjacency/laplacian/distance/ distance path/augmented vertex degree/extended adjacency/vertex connectivity/ random walk markov/weighted struct func 1/weighted struct func 2) |

### Edge-Weighted Properties

| Node-Level: Strength, Assortativity, Disparity, Geometric Mean of Triangles, Edge-Weighted Local Clustering Coefficient, Edge-Weighted Interconnectivity | Edge-weighted clustering coefficient identified the gene modules in co-expression network\(^\text{45}\). Edge-weighted interconnectivity ranked the candidate disease genes in biological networks\(^\text{16}\). |
| Network-Level: Weighted Transitivity, Edge-Weighted Global Clustering Coeff |

### Node-Weighted Properties

| Node-Level: Node Weight, Node-Weighted Local Clustering Coeff, Node-Weighted Neighbourhood Score | Node-weighted neighbourhood score prioritized the novel disease genes for the prediction of drug targets for a given disease\(^\text{15}\). |
| Network-Level: Total Node Weight, Node-Weighted Global Clustering Coefficient |

### Directed Properties

| Node-Level: In-Degree, In-Degree Centrality, Out-Degree, Out-Degree Centrality, Directed Local Clustering Coefficient, Neighbourhood Connectivity (in/out/in-&-out), Average Directed Neighbour Degree | In/out-degree, and clustering coefficient analyzed the gene regulatory networks under different conditions\(^\text{111}\). Directed clustering coefficient and average directed neighbour degree studied the neuro-connectivity networks\(^\text{8}\). |
| Network-Level: In-Degree (max, avg, min), Out-Degree (max, avg, min), Directed Global Clustering Coefficient |

---

Table 12 (continue) | List of network descriptors (node-level, network-level, and edge-level) in different categories provided by PROFEAT and the selected systems biological applications
(G) Reference

55 Li, X. & Gutman, I. *Mathematical Aspects of Randic-Type Molecular Structure Descriptors*. (University of Kragujevac, 2006).
60 Gutman, I. *Graph theory and molecular orbitals. XII. Acyclic polyenes*. The *Journal of Chemical Physics*, 62, 3399, (1975).


