New Features and Tutorial

Graphvar 2.02
added dynamic network measures as in Seizemore et al., 2017
added community functionalities as in Fornito et al., 2012
What is this new feature?

It can be used to obtain a group based consensus of the graph decomposition

"... it's like a group ICA on graphs"
Community functionalities as in 

Fornito et al., 2012

Tutorial
Computes a *consensus community structure* across all subjects.

Computes *consensus community structures* for two groups and compares them.

Resolution to cluster the graph:
- gamma>1: detects smaller modules
- 0<=gamma<1: detects larger modules
- gamma=1: classic modularity (default)

Implemented procedures work with fully connected matrices and with thresholded graphs.

* as implemented in the BCT

added community functionalities as in e.g. [Fornito et al., 2012](#)
Consensus Community structure - general procedure (text adapted from Dwyer et al. 2014):

1. **Individual level modular decomposition:**
   We run multiple iterations (1000) of the Louvain/Newman modularity algorithm (which is an optimization algorithm and thus produces slightly different outcomes per iteration) to obtain a set of possible clusters in the graph. To identify the final clustering solution, we use a consensus-based approach in which we generate a co-classification matrix (in which each \([i, j]\) element contained 1 if two nodes were classified in the same module and 0 otherwise) and subsequently run a second decomposition of this co-classification matrix (c.f. Lancichinetti and Fortunato, 2012). In this manner, nodes frequently co-classified in the same module across multiple iterations of the algorithm will be assigned to the same module in the final solution.

2. **Group-level modular decomposition:**
   To obtain a group based consensus of the graph decomposition, we pass the final consensus decompositions for each individual to a second level to derive a group-level representation of network modularity based on a similar logic to the consensus approach used at the single-subject level. Specifically, the individuals final consensus decompositions are summed across individuals to generate a sample-level consistency matrix. A high weight in elements of this matrix indicates that two nodes were frequently classified in the same module across individuals. As such, a subsequent modular decomposition of this group-level consistency matrix ensures that nodes frequently co-classified together are likely to be assigned to the same module in the final solution.

By aggregating results across single-participant decompositions, this consensus-based approach allows to derive a group-level representation of community structure while also characterizing interindividual variability in network organization using classification consistency and diversity metrics. Such analyses are not possible when decomposition is performed on a group-averaged correlation matrix. Consensus-based approaches have also been shown to yield more stable individual module solutions, given the known degeneracy of most graph theoretic module detection algorithms (Good et al., 2010; Lancichinetti and Fortunato, 2012).
All subjects:
(Variable sheet with Subject IDs required)

Performs a group consensus clustering across all subjects that are loaded in the current workspace.
Two groups:
(Variable sheet with Subject IDs required)

TESTING BETWEEN-GROUP DIFFERENCES IN MODULAR ORGANIZATION:

To evaluate the significance of between-group differences in modular organization we use a permutation approach. By permuting labels across groups and re-calculating the difference between groups in the respective modularity metric (see next slide) we obtain a distribution of this group difference under the null-hypothesis. By placing the real group difference “delta” in the random distribution of deltas we can determine the significance from its percentile position in the distribution.
If you have performed previous modularity analyses on different sets of subjects with the “All subjects” function, you can compare the results by loading the respective “GroupCommunity” folder (similar to “Two groups” function).
Measures of modular organization I:

**OPTIMAL MODULAR DECOMPOSITION – Q** (text adapted from Fischi-Gomez et al., 2016):

In the Louvain modularity algorithm, Q is obtained by iteratively repeating 2 steps until convergence to a modularity maximum (Q).

First, each node is placed in a separate module, and all possible node moves between modules are evaluated in terms of modularity gain (step 1). When no individual move can further improve the Q value, nodes belonging to the same community are agglomerated (step 2) in order to form new 'super-nodes'. Step one (moves evaluation) is repeated on the new 'super-nodes' network. The two steps are repeated until convergence.

Q is always calculated (no selection needed)
CLASSIFICATION CONSISTENCY AND DIVERSITY (text adapted from Fornito et al., 2012):

To understand the functional roles played by each module and their constituent nodes, one can examine the consistency and diversity with which different regions are coclassified into the same module across participants.

Classification consistency is estimated by computing the within-module strength, $z$, of each node in the group-level consistency matrix. Classification diversity is computed using the diversity coefficient $h$.

Applied in this context, $z$ quantifies the degree to which each region is classified in the same module across participants relative to other nodes in the same module. Brain regions with high $z$ values represent core components of their module and thus act as local connectivity hubs. The diversity coefficient, $h$, quantifies the variability of each region’s modular assignment across participants. Regions with high $h$ have a relatively equal probability of being classified into different modules across participants, because their connectivity is dispersed between modules from individual to individual. These regions, therefore, represent transitional nodes that facilitate functional integration between modules.
Measures of modular organization III:

**PARTITION DISTANCE** (text adapted from Fischi-Gomez et al., 2016):

Quantifies the distance between pairs of community partitions with information theoretic measures: mutual information and variational information (Meila, 2007).

These two measures, based on the concept of entropy, quantify similarities and differences between graphs partitions. The **mutual information (MI)** quantifies how much information is shared by the two (different) partitions $C_i$ and $C_j$ of a given network $G$. Roughly speaking, MI tells how much we learn about $C_i$ if we know $C_j$, and viceversa. Nevertheless the most commonly used measure of similarity in graph is the normalized mutual information ($\text{MIn}$), introduced by (Danon et al., 2005). This measure equals 1 if the two partitions are identical, whereas it has an expected value of 0 is the two partitions are independent.

The **variation of information (VI)** expresses the quantity of information intrinsic to the two partitions, corrected by the information shared by the two partitions. VI is upper-bounded by the logarithm of the number of nodes ($\log n$) and can be therefore normalized by this value, giving a rescaled value of VI to the range [0,1].
Output visualization:
Saved output in folder „Group Community“:

- Binarized Affiliation vectors (one per module)
- Affiliation vector (assignment of nodes to modules)
- Group-level consistency matrix
- Classification diversity (one value per node)
- Modularity quality index Q*
- Classification consistency (one value per node)
- Affiliation vector as table

Figures created by BrainNetViewer and zh-Plots are also saved in this folder
Saved output in folder „GraphVars“:

- Difference in h between groups
- Difference in Q between groups
- Difference in Z between groups
- MIn – between groups
- VIn – between groups
- Subject-level consistency matrices (one per subject)
- Permutation distribution of difference in h (sorted)
- Permutation distribution of difference in Q (sorted)
- Permutation distribution of difference in Z (sorted)
- Permutation distribution MIn (sorted)
- Permutation distribution VIn (sorted)
- Permutation generated Affiliation vectors of rand groups
- Permutation generated h per region per permutation
- Permutation generated Q per permutation
- Permutation generated Z per permutation

Subject-level consensus affiliation vector (one per subj)
- Overlap of nodes in modules: Group 1 -> Group 2
- Overlap of nodes in modules: Group 2 -> Group 1
- P-value for h (one per region)
- P-value for MIn
- P-value for Q
- P-value for VIn
- P-value for Z (one per region)
Simply use (one of) the binary affiliation vectors as input to the BrainRegions xlsx sheet (first column) for subnetwork analyses