


SOFTWARE

Open Access



An additional k-means clustering step improves the biological features of WGCNA gene co-expression networks

Juan A. Botia^{1,2*} , Jana Vandrovcova², Paola Forabosco³, Sebastian Guelfi², Karishma D'Sa^{1,2}, The United Kingdom Brain Expression Consortium, John Hardy², Cathryn M. Lewis¹, Mina Ryten^{1,2} and Michael E. Weale¹

Abstract

Background: Weighted Gene Co-expression Network Analysis (WGCNA) is a widely used R software package for the generation of gene co-expression networks (GCN). WGCNA generates both a GCN and a derived partitioning of clusters of genes (modules). We propose k-means clustering as an additional processing step to conventional WGCNA, which we have implemented in the R package *km2gcn* (k-means to gene co-expression network, <https://github.com/juanbot/km2gcn>).

Results: We assessed our method on networks created from UKBEC data (10 different human brain tissues), on networks created from GTEx data (42 human tissues, including 13 brain tissues), and on simulated networks derived from GTEx data. We observed substantially improved module properties, including: (1) few or zero misplaced genes; (2) increased counts of replicable clusters in alternate tissues (x3.1 on average); (3) improved enrichment of Gene Ontology terms (seen in 48/52 GCNs) (4) improved cell type enrichment signals (seen in 21/23 brain GCNs); and (5) more accurate partitions in simulated data according to a range of similarity indices.

Conclusions: The results obtained from our investigations indicate that our k-means method, applied as an adjunct to standard WGCNA, results in better network partitions. These improved partitions enable more fruitful downstream analyses, as gene modules are more biologically meaningful.

Keywords: Gene co-expression networks on brain, K-means applied to WGCNA, Assessment of better gene clusters on bulk tissue

Background

Systems biology is a descriptive paradigm in which one of the main concerns is how genes work together to form subsystems. A basic assumption within this context is that genes which are co-expressed are often in the same subsystem [1]. Gene co-expression networks (GCN) are graph-based models used to express such subsystems. Construction of these networks is usually based on co-variation in expression within groups of genes across samples [2]. They are graphs in which nodes are genes and edges represent interactions between them. Typically,

the edges are undirected, in the sense that causality (e.g. whether changes in Gene A expression causes changes in Gene B expression) is unassigned. Edges may be weighted and/or signed, thus indicating the strength of relationship between pairs of genes and up/down regulated interactions depending on the sign. Topological considerations, such as the number or relevance of connections for each node, can distinguish some nodes as highly interconnected (hubs) and central nodes within the system being modelled.

GCN can be used to make *in silico* functional predictions about genes. The Guilt By Association (GBA) paradigm [3] is used to predict function for genes that are not sufficiently studied and annotated using GCNs. GBA assumes that genes that strongly co-express must share functionality, thus we can use well-characterised genes to assign function to those that are not.

*Correspondence: j.botia@ucl.ac.uk

¹Department of Molecular Neuroscience, Institute of Neurology, University College London, Queen Square, WC1N 1NP London, UK

²Department of Medical & Molecular Genetics, School of Medical Sciences, King's College London, Guy's Hospital, SE1 9RT London, UK

Full list of author information is available at the end of the article

Groups of genes that tightly co-express are usually seen as a single functional unit. On this basis, in the same way that single genes are used in association mapping with phenotype, convenient mathematical representations of groups of genes can be useful for multi-gene association mapping with phenotype as well. One of the most widely used pipelines for GCN construction is Weighted Gene Co-expression Network Analysis (WGCNA) [4–6]. It works in two main steps. In the first step it constructs a network N of gene-gene co-expression in the form of a squared $n \times n$ matrix, where n is the number of genes in the study and each $N(i, j)$ is the interaction strength between the corresponding pair of genes (i.e. adjacency). In the second step, this matrix is used as the basis for obtaining a new squared distance matrix with the distance between genes, ready to be used for obtaining clusters. And then such clusters can be used for multi-gene association mapping with traits or different downstream analyses [7–9]. This pipeline has been widely used and generated many fruitful insights into how genes interact within specific conditions [7–13].

In this paper we propose an improvement to the standard WGCNA pipeline by a refinement of how the clusters (i.e. modules) are generated. This refinement is enabled through a hybrid clustering algorithm. It uses the output of the conventional WGCNA clustering as subsequent input to a k-means [14] clustering algorithm for further refinement. We will show that this hybrid scheme improves many interesting module properties paving the way to more accurate and potentially useful WGCNA co-expression network analyses.

WGCNA's standard configuration uses hierarchical clustering (HC). In HC, a strong point is that the dendrogram structure eases the problem of finding a good number of clusters, k . Moreover, the developers of WGCNA include in the software an automated method to generate the appropriate number of clusters [15]. On the other hand, a weak point of HC is that final results strongly depend upon how distances between clusters are compared. Furthermore, once the decision on which branch of the dendrogram a gene belongs to, this cannot be undone.

Regarding k-means, a weakness in it is that the value of k (i.e. number of clusters) must be set prior to running the algorithm. Although there are techniques for setting it automatically, most of these are based on multiple random initialisations of centroids (e.g. k-means++ [16]), so k is usually set arbitrarily. It needs an initialisation of the centroids to start running. A centroid is defined as an average representative of all the genes/points within the cluster such that all genes/points belonging to the cluster show minimum distance to that centroid in comparison to the other modules. How we initialize these centroid will have a critical effect on the final result. On the upside, k-means will search for the best centroids quickly and will quickly

converge to an equilibrium situation (see “Improvement of hierarchical clustering with k-means” section).

The hybrid scheme we propose exploits the upsides from both approaches while alleviating their respective drawbacks. K-means will move genes between modules thus effectively undoing premature decisions made by HC when assigning genes to sub-dendrograms. We set the value of k equal to the number of modules discovered by HC and we initialise the centroids to the eigengenes generated by WGCNA, thus taking advantage of HC to carry out sensible initialization (see “The standard WGCNA procedure” section).

Implementation

The standard WGCNA procedure

Consider a gene expression profile matrix $G_{n \times m}$ where n is the number of samples for a given condition, m is the number of transcripts and each $g(i, j)$ in G gives the quantification of the j -th transcript within the i -th sample. The standard WGCNA [6] procedure generates a squared adjacency matrix, between genes, based on their correlation. Depending on whether the adjacency is signed (where correlations in the $[-1, 1]$ interval are scaled into the $[0, 1]$ interval) or unsigned (where negative correlations are made positive) we will obtain networks either reflecting the direction of co-regulation (i.e. up or down regulation) or ignoring it, respectively. Adjacency is defined as $adj(i, j) = |cor(i, j)|^\beta$ for genes i and j . The β parameter is an integer that modulates how smooth is the transition between the lowest to the highest possible co-regulation between genes.

The WGCNA methodology enables choosing β in such a way that the network shows a Scale Free Topology (SFT) property [17] (where the network has the same shape whether ‘zoomed-out’ or ‘zoomed-in’). This feature is commonly observed in biological networks. From the adjacency values, a new matrix with the same dimensions is created, the Topological Overlap Matrix (TOM). This step alleviates the effect of noisy genes when obtaining the adjacency from correlation.

Once the network is built through the TOM, it is converted to a distance matrix ($1 - TOM$) to use it as the basis for clustering (HM with average linkage distance comparison between clusters). A dynamic tree-cutting algorithm [15] is then applied to the dendrogram to generate a partition $P = \{P_1, \dots, P_k\}$ of disjunct sets of genes.

Thus, WGCNA generates two main components which are useful for subsequent downstream analyses. On the one hand, the TOM gives, for the j -th row/column, the level of co-expression of gene j with all of the genes in the network. The higher the value for a given (i, j) pair, the tighter the interaction between them. Furthermore, the sum of all row or column values for a gene, will give a measure of its overall level of co-regulation within the

experimental condition, i.e. its ‘hubiness’. Thus, the TOM is, in effect, the GCN.

The other component produced by WGCNA is the partition of gene sets, P , created from the TOM. These partitions or modules often reflect cell types, common cellular functions or other biological subsystems reflecting, for example, immune function, or function related to the tissue under study [2, 7, 8]. But the main utility of modules is to allow mapping gene groups to traits, when available. Following the WGCNA standard methodology, this is performed by looking for significant correlations between traits and the module ‘eigengene’. The eigengene summarizes the overall module activity in a given sample, and is obtained as the 1st PC component of the gene expression of genes belonging to the module.

Improvement of hierarchical clustering with k-means

HC provides a convenient graphical representation of groupings that can be validated by biologists. One can readily obtain a suitable number of clusters from such an approach by ‘cutting’ the dendrogram at different heights, either manually or via various automatic algorithms [15]. But, as we explained above, the final dendrogram strongly depends on how we measure distance between clusters (e.g. via simple, complete or average linkage). Furthermore, once a gene falls under a subdendrogram, this decision cannot be modified under HC.

If we consider how WGCNA manages modules and eigengenes, it is assumed that each gene is highly correlated with other genes in its module. In other words, the module membership (MM) of the gene in its own module, measured as the Pearson correlation between its expression and the module eigengene, should be higher than it is for any other module. However, we show here, from our real-data analyses, that 25% of genes would be better off in other modules (see “K-means improves the ‘eigengene’ as a tool for analysis” section).

In this paper, we propose a post-processing step based on k-means to overcome all these limitations. It works on the partition P , leaving the TOM unmodified. The k-means algorithm [14] is well known and works on the n dimensional sample space of m points (genes) in an iterative fashion. It starts by setting a value for k , the number of clusters to discover and k centroids, one for each cluster. Centroids are the representatives of each cluster, in such a way that a point (gene) g belongs to cluster i if the distance of such point to the cluster centroid is the minimum among all distances to all k cluster centroids. In standard k-means, given a partition of k modules, the the centroid for the i -th module $c_i = \{c_{i,1}, \dots, c_{i,n}\}$ is generated as follows

$$c_i = \frac{1}{n} \sum_{j=1}^m g_j, \text{ where } g_i \in p_i. \quad (1)$$

However, in WGCNA, the notion of a centroid is substituted by that of an eigengene. Accordingly, our definition of k-means will use eigengenes as centroids.

The concept of distance is a central element of k-means. It is important to note that distance in k-means is always defined between a point in the dataset (i.e. a gene) and a centroid (i.e. an eigengene). Euclidean distance is the the most commonly used distance in conventional k-means. However, given that we are constructing co-expression networks based on correlations, distance cannot be Euclidean. Modules should represent co-expressed genes (i.e. highly correlated) instead. Thus, and depending on the WGCNA type of network, we should apply a distance between gene and eigengene based on the co-expression measure used. We will limit our discussion to signed networks. These specific types of networks will separate up- from down-regulated genes in different modules, which is usually of biological interest. They are also convenient for downstream analysis as correlation of genes and eigengenes will be positive, which eases a posteriori analyses. In signed networks, WGCNA uses

$$co(g_i, eg_j) = \frac{1}{2}(1 + cor(g_i, eg_j)), \quad (2)$$

as a normalised measure of co-expression between the expression profile of a gene g_i and a eigengene eg_j , where by default $cor()$ is the Pearson correlation coefficient. Accordingly, we use $1 - co(g_i, eg_j)$ as distance. It is worth noting that HC needs a distance matrix between all genes, i.e. $1 - TOM$. K-means needs instead a computable distance definition between gene and eigengene. Finally, on the basis of this definition of centroid and distance, genes are reassigned to the partitions induced by the new centroids, iteratively. If a stopping criterion is met, the algorithm finishes. Otherwise, a new iteration is performed.

We note that WGCNA is computationally optimized to use Pearson correlation. Other correlation measures are in principle possible, including Spearman’s rank correlation coefficient. However, in our own investigations we observe an increase in computation time of at least $\times 2.5$ when using Spearman correlation without seeing any conclusive improvement with respect to the biology of networks (data not shown). Thus, throughout this paper we perform analyses using Pearson correlation.

We propose a general procedure which obtains, from a $G_{n \times m}$ matrix of gene expression profiles from n samples and m genes, a clustering partition P of such genes by incorporating the standard WGCNA process together with a post-processing of the partition obtained from it.

The original contribution of this paper is described in steps from 5 to 8 below.

- Step 1: Initialization. Let $G_{n \times m}$ be a dataset of n samples and m genes for a given condition. Let $d()$ denote a distance function between a gene in G and an eigengene. Let $f_c()$ denote a function which takes a clustering partition $P = \{p_1, \dots, p_k\}$ as an argument and generates centroids (i.e. k vectors, one for each p_i , of n components)
- Step 2: $\beta =$

```
WGCNA::pickSoftThreshold(data=G,
powerVector=1:20,
networkType='signed')$powerEstimate
```
- Step 3: Obtain a TOM, given G and β
- Step 4: Generate a partition $P_{HC} = \{p_1, \dots, p_k\}$ with $1 - TOM$ as a distance matrix and with average linkage hierarchical clustering and dynamic cutting height.
- Step 5: Let $c = \{c_1, \dots, c_k\}$ be a set of k vectors of n components which denote the centroids of the k-means clustering.
- Step 6: Initialize c with $f_c(P_{HC})$
- Step 7: Create a new partition $P_{kM} = \{p_1, \dots, p_k\}$ by assigning each gene g_i , $1 \leq i \leq m$ to a $p_j \in P_{kM}$ such that $d(g_i, c_j) \leq d(g_i, c_t)$, $1 \leq t \leq k$ holds.
- Step 8: If the termination criterion holds, then STOP. If not, generate new centroids c with $f_c(P_{kM})$ and Go to Step 7.

Note that, in this algorithm, we left $d()$ and $f_c()$ undefined. However, within this paper we define $d()$ according to Eq. 2 and $f_c()$ as the module eigengene.

Computational complexity of the proposed approach

Conventional WGCNA GCN construction needs three sequential steps: (1) obtaining the soft threshold (i.e. β parameter) to account for scale free topology, which has a computational complexity that depends on the number of genes and samples, (2) obtaining the TOM matrix, which has a complexity $O(n^2)$, where n is the number of genes, as it has to construct a $n \times n$ squared matrix of adjacencies between all n genes, and (3) hierarchical clustering, which has a complexity $O(n \log n)$. Overall, WGCNA's computational complexity is $O(n^2)$. WGCNA's space complexity is also $O(n^2)$ because it needs to maintain the TOM in memory for HC to get the clusters. The computational complexity of k-means fits well with WGCNA's complexity. Its time complexity is $O(n \times k \times it)$ where n is the number of genes, k is the number of clusters and it is the number of iterations. Assuming that $k, it < 100$, using it as a post-process is very affordable in terms of computation time. Note that k-means does not require the TOM matrix in memory as the only distances it requires are between

genes and eigengenes, and these we obtain on the fly by using Eq. 2.

Stopping criterion

It is reasonable to assume that a sufficiently high number of k-means iterations will always be able to decrease the number of misplaced genes (i.e. genes which lie closer to the centroid of a different module) to 0. On the other hand, the algorithm's time complexity (see "Computational complexity of the proposed approach" section) means that it is possible to run a single k-means in a conventional laptop in a matter of a few minutes. This means that we could, in principle, design a stopping criterion based on the minimum number of misplaced genes being set to 0. However, we note that a situation may exist where the algorithm may fall into an infinite loop without reaching the desired state (i.e. changing the same genes from one module to another and back again). Thus, the stopping criterion we include in the software package `km2gcn` tries to reach the desired value for misplaced genes but always within a limited number of iterations. We did not observe the mentioned *infinite loop* situation in any of our experiments.

Results

We wished to assess the ability of our method to define gene groups that genuinely reflect biological function. This is non-trivial for the following reasons. Firstly, many genes are known to be pleiotropic, i.e. a single gene can affect many traits [18]. Transcription factors are a good example of this [19] but there are many other examples [20]. By creating non-overlapping partitions we deliberately ignore this fact and implicitly assume a model in which genes are highly specialized (i.e. belong to a single module). Secondly, we are limited by technology and sample availability from producing optimum estimates of gene expression profiles. We therefore lack of all the necessary information to build the best model. Finally, if we wanted to evaluate the functional similarity of genes within a module, again we do not know all functions that all genes may play in any condition.

Notwithstanding these caveats, we explored various approaches to provide a comprehensive and varied assessment of the effectiveness of our k-means hybrid method. In "Materials and methods for the GCNs used for our evaluations" section we describe the datasets used in our investigations and the particular pipelines used to obtain the corresponding GNCs. In "Dynamics of k-means when working on 1-TOM distance space" section we show our hybrid approach (i.e. the combination of HC and k-means) works. In "Is k-means doing a proper job?" section we digress to note that k-means actually optimizes the sum of squares of within cluster distance. In "K-means improves the 'eigengene' as a tool for analysis" section we show

that the proposed approach improves modules as a tool for mapping with traits. In “K-means improves module preservation” section we suggest that k-means improves cluster similarity between conditions (i.e. tissues in this case). In “K-means detects more accurate partitions than WGCNA in simulated data” section we compare the accuracy of k-means against WGCNA on simulated data. In “K-means improves functional enrichments” section we show that k-means improves a module’s functional characterization through well-known databases such as the Gene Ontology. Finally, in “K-means improves brain specific cell type marker enrichment” section we present results that suggest that gene markers for specific cell types show a better arrangement in partitions generated from k-means.

Materials and methods for the GCNs used for our evaluations

We evaluated GCNs in two well-known datasets. The first (the United Kingdom Brain Expression Consortium or UKBEC dataset) is focused on brain tissue exclusively and it is based on Affymetrix Human Exon v2 microarray expression profiles from 10 brain tissues. This dataset is well suited for evaluating the k-means extension to WGCNA because it is well known, it comprises 10 different brain regions and GCN networks created with the standard WGCNA method have been published [8]. The procedure used to create the GCNs is as follows. Sample outliers were identified by visual inspection after clustering the samples using hierarchical clustering with Euclidean distance as the distance measure. The majority of the identified outliers had low interarray correlation, which is defined as the Pearson correlation coefficient of the expression levels for a given pair of transcripts using all available data available (i.e., < 3 standard deviations of the average interarray correlation). After outlier removal, the same process was repeated to check for additional outliers. The GCN constructed was of signed type, with $\beta = 12$ for all tissues. Using these settings, the HC WGCNA partition was created using 15,409 transcripts (13,706 genes) passing quality control. Once the partition was created, 3743 additional transcripts (3541 genes) were assigned to modules based on their highest module membership. Each partition was refined afterwards with k-means.

The second dataset is GTEx [21], which is one of most comprehensive human datasets currently available for multi-tissue transcriptomics. The GTEx V6 gene expression dataset comprises 11,978 samples unevenly distributed across 54 post-mortem human tissues. We created networks for 42 tissues. In sequential steps, starting from RPKM [22] values of gene-level quantification provided by GTEx, we selected all tissues with more than 60 samples. For each tissue we retained Ensembl genes

with $RPKM > 0.1$ seen in more than 80% of the samples. This produced a variable set of genes for each tissue, with a minimum of 16,098 for skeletal muscle tissue and a maximum of 29,561 for testis. We applied batch, gender, age and RIN as known covariates for data correction and to account for unknown covariates we applied SVA (surrogate variable analysis)[23] axes. For each dataset of filtered RPKM values, we applied the *sva* R package using `svaseq()` to generate SVA axes. For network construction we used the residuals obtained by regressing the RPKM expression values with the known and unknown covariates with a generalized linear model. To construct the networks, we applied the algorithm introduced in “Improvement of hierarchical clustering with k-means” section.

Note that the differences between the UKBEC and GTEx networks are important and makes them well suited and complementary for the purpose of our study. The UKBEC gene expression dataset is microarray based, while the GTEx gene expression dataset is based on RNA-seq technology, with RPKM quantification. UKBEC networks are restricted to 10 brain tissues while GTEx networks cover 42 tissues, including 13 brain tissues (see Additional file 1 for tissues used, number of samples and genes). In summary, we have 42 GTEx GCNs, 10 brain specific UKBEC GCNs, GTEx sample sizes range from $n = 63$ to $n = 430$ (mean 182); UKBEC sample sizes range from $n = 65$ to $n = 88$ (mean 78.8); we have a variable number of genes used in the GTEx GCNs, in the range 16,098 to 29,561 for skeletal muscle and testis respectively (mean 19,636); and the same 19,152 probes for all 10 UKBEC GCNs. Finally, note there is a much higher variability in the number of modules per GCN in GTEx, [10, 214] (mean 67.6) than in UKBEC, [13, 34] (mean 22).

Please note that throughout the paper we use abbreviations to refer to tissues. Please see Table 1 for the correspondence between abbreviations and brain region names.

Dynamics of k-means when working on 1-TOM distance space

As outlined in “Improvement of hierarchical clustering with k-means” section, our proposed algorithm does not modify the distance matrix (i.e. $1 - TOM$) but acts later, on the partition $P = \{p_1, \dots, p_k\}$ taking k from the number of modules discovered by the HC used within WGCNA. K-means acts iteratively creating centroids from modules and deciding for each gene, on the basis of the new centroids, which one is nearest to the gene. If, in the current iteration, the gene is nearest to a different centroid, then the k-means algorithm assigns it to the corresponding module. Thus, in each iteration a new partition is generated with the changes applied to the former partition.

Table 1 Real names for the tissues used in the UKBEC and GTEx brain tissue experiments

Short name	UKBEC Tissue name	Samples	Short name	GTEx Tissue name	Samples
CRBL	Cerebellum	76	AMYG	Amygdala	72
FCTX	Frontal Cortex	83	ACCT	Anterior cingulate cortex (BA24)	84
HIPP	Hippocampus	86	CAUD	Caudate (basal ganglia)	117
MEDU	Medulla	88	CEHE	Cerebellar Hemisphere	105
OCTX	Occipital cortex	77	CERE	Cerebellum	125
PUTM	Putamen	77	CTEX	Cortex	114
SNIG	Substantia nigra	65	FCTX	Frontal Cortex (BA9)	108
TCTX	Temporal Cortex	72	HIPP	Hippocampus	94
THAL	Thalamus	81	HYPO	Hypothalamus	96
WHMT	White matter	83	NUAC	Nucleus accumbens (basal ganglia)	113
			PUTM	Putamen	97
			SPIN	Spinal Cord	71
			SNIG	Substantianigra	63

Figure 1 displays the dynamics of the algorithm in terms of how genes are changed from one module to another. In all analyses displayed there is a high activity in terms of moved genes in the early iterations, which progressively decreases to reach a stable level of changes close to zero. The number of changes at the first iteration ranges roughly between 3000 and 5000 genes, i.e. about 1/4 of the gene pool size. Any single gene can be moved more than once during the series of iterations (for more details on gene changes and how the algorithm stabilizes see “Is k-means doing a proper job?” section). It is also of interest to note that multiple modules contribute to the final configuration of genes to each other module. For example, with the 42 GTEx GCNs, for each module $p_i \in P$, on average 30% of other modules within the GCN contribute with genes to its final gene set configuration. Genes that leave their HC module have a module membership at that module of 0.53 on average, with standard deviation of 0.19. Genes arriving to a module for the final k-means partition show an average MM on arrival, 0.57, with standard deviation 0.18.

The lower panel of Fig. 1 focuses on the 10 UKBEC GCNs, and on how MM evolves with iterations. Dashed lines show, for each tissue, the average MM of moved genes, at each iteration, defining MM with reference to the original WGCNA partition. Initially, the algorithm focuses on moving genes with very low MM, but following this it focuses on genes with higher MM and then stabilizes. The solid lines show the average MM of all the genes in the network for each iteration. This dramatically increases over the first iterations and then smoothly and monotonically increases across additional iterations. This suggests that, over time, genes’ assignment to a module becomes stronger.

Is k-means doing a proper job?

k-means is designed to optimize the sum of squares of within clusters distance [14]. We define the within cluster distance, denoted with $W(P)$, for a partition P as

$$W(P) = \sum_{k=1}^K \sum_{p(i)=k} \|g_i - c_k\|^2 \quad (3)$$

where K is the number of clusters within P , $p(i)$ refers to the cluster that g_i belongs to, and c_k is the eigengene for the k -th cluster. We could, alternatively, define the distance between clusters, $B(P)$. Note that for a given set of genes, we can obtain the sum of distances between all gene pairs. If we denote this measure by $T(G)$ for a given gene expression profile G , we can decompose it into $T(G) = B(P) + W(P)$ for any given P obtained from G (see [24] for a detailed discussion). This means that maximizing the between-clusters distance is equivalent to minimizing the within-cluster distance.

We observe that k-means monotonically decreases $W(P)$ (and thus $B(P)$ increases) across iterations (see Fig. 2). The algorithm generates a higher $W(P)$ at the early iterations, which decreases to a lower level in later iterations. This behaviour is in line with the shape of the gene changes curves of the upper plot in Fig. 1. Higher number of moved genes imply higher decreasing rate of within cluster distance.

This behaviour is also in accordance with what we see in Fig. 3. This plot shows, for each module $p_i \in P$, and the specific case of UKBEC’s cerebellum CGN, the distance between the eigengenes for the same module, as they are created during successive iterations. Over time, the eigengene vectors stabilize across iterations suggesting that cluster definition becomes stable.

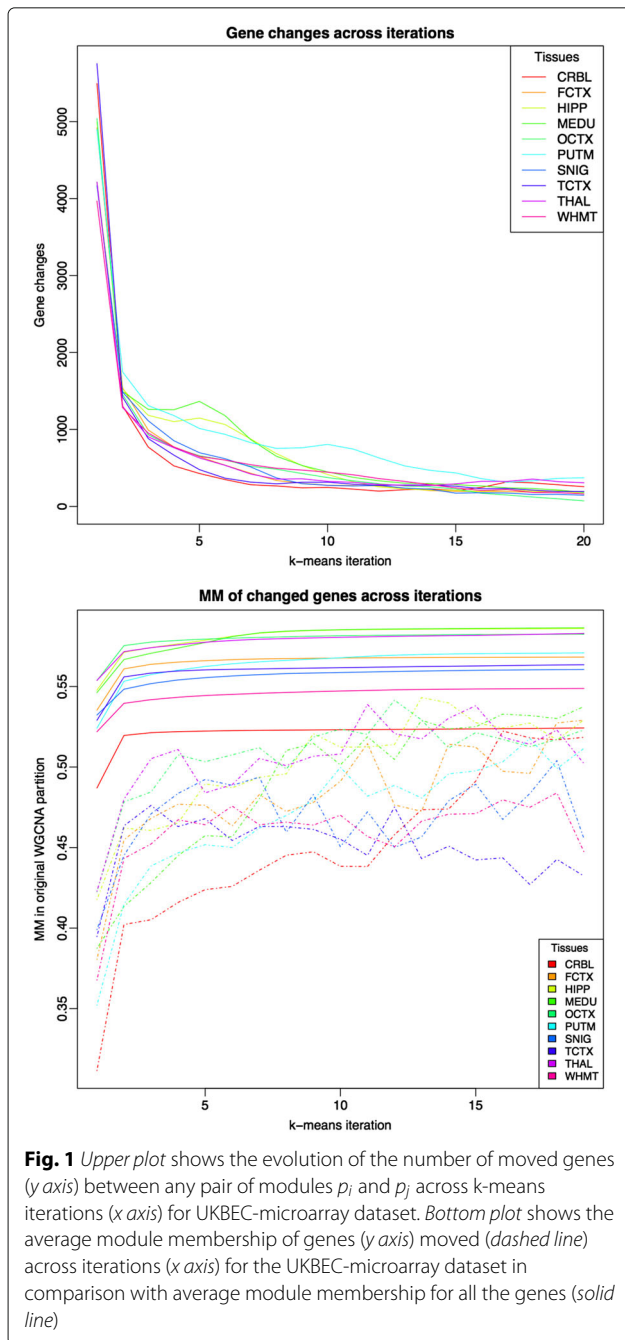


Fig. 1 Upper plot shows the evolution of the number of moved genes (y axis) between any pair of modules p_i and p_j across k-means iterations (x axis) for UKBEC-microarray dataset. Bottom plot shows the average module membership of genes (y axis) moved (dashed line) across iterations (x axis) for the UKBEC-microarray dataset in comparison with average module membership for all the genes (solid line)

K-means improves the ‘eigengene’ as a tool for analysis

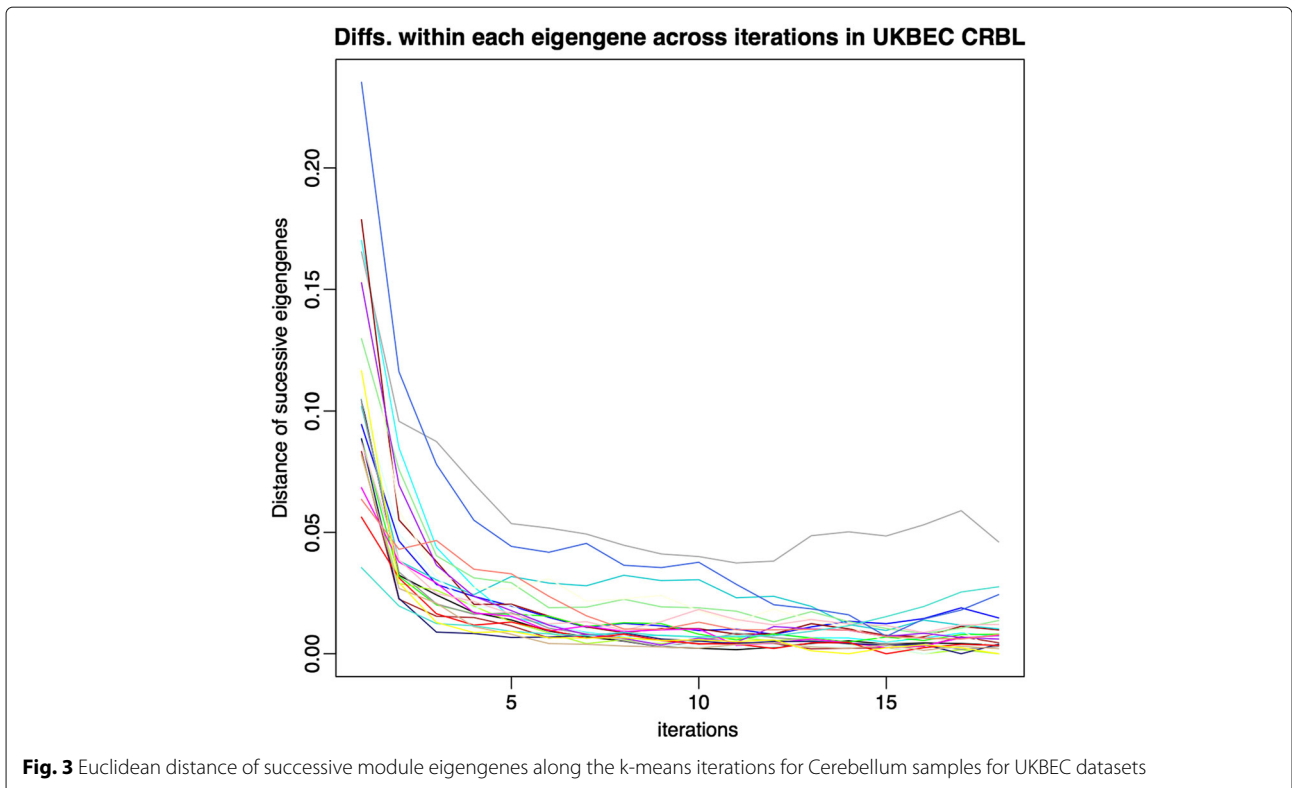
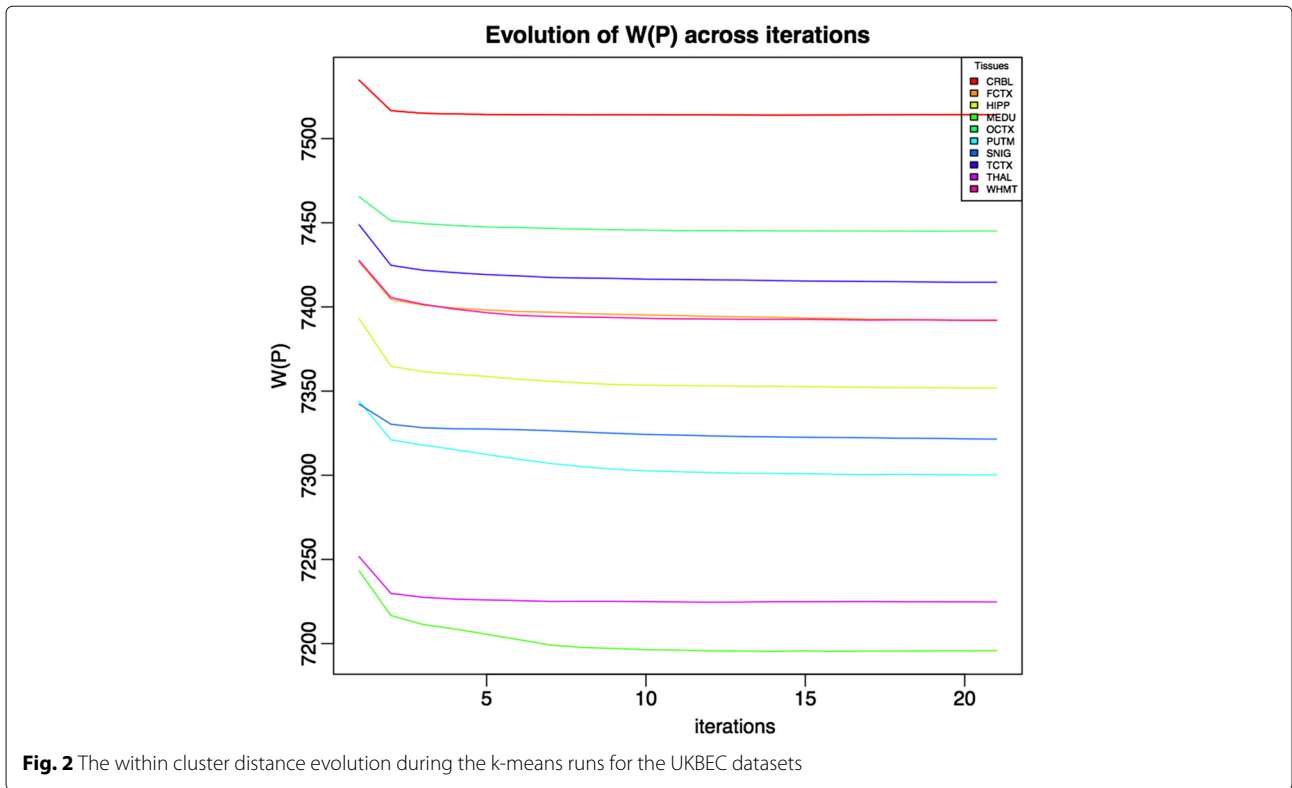
One of the main applications of WGCNA partitions is searching for associations between gene clusters and traits. Traits are usually given as a vector of n components where n is the number of samples. On the other hand, modules are comprised of m vectors of the same form where each vector is the expression profile for the corresponding gene. To assess the correlation between the trait and the gene module, WGCNA transforms module p_i into an eigengene (i.e. the 1st principal component of

gene expression). From this, the correlation between the trait and the eigengene can be easily obtained.

There are several applications for the eigengene. For example, it can be used to provide a measure of how strong is the membership of each gene $g \in p_i$ to the i -th module, by correlating its expression with the eigengene, resulting in the MM of g . Let this module membership be denoted with $m(g, i)$ for gene g and module i . It is assumed that a good P would be one such that, the number of genes g with $m(g, i) < m(g, j)$ when $g \in p_i$, for any $i \neq j$, is low. Let us call such genes ‘misplaced genes’. We would prefer a partition in which the number of misplaced genes is minimum. To assess the number of misplaced genes after applying k-means, we performed investigations in the 10 UKBEC and 42 GTEx tissues. In UKBEC, using k-means with only 20 iterations (i.e. the fixed number of iterations used in all the experiments), we get a maximum of 380 misplaced genes in the putamen and a minimum of 72 for occipital cortex and a mean of 208 misplaced genes per partition. In the WGCNA partitions, the maximum number of misplaced genes is 5742 in temporal cortex, the minimum is 3970 in white matter and an average of 4763 genes; 20 times more than with the k-means algorithm. In GTEx clustering partition modules, the average number of misplaced genes in modules from a WGCNA partition is 118. After applying k-means, it is only 0.4.

K-means improves module preservation

One component of WGCNA provides a convenient tool for the analysis of module preservation [5]. Given a partition, P , constructed from a network obtained from a given set of samples S , we can test whether the features of each module $p_i \in P$ (i.e. cluster and network based features) are preserved in an alternative set of samples S' (e.g. a different species but same brain region, or same species but different brain region). Preservation analysis is based on estimating, for some statistic of interest, differences between what is observed and what is obtained by random permutation. For example, one statistic of interest is the gene correlation with the eigengene (kME). Through a simple transformation one can check whether the values obtained in the reference network are maintained (i.e. correlated) for the same genes within the other network. WGCNA uses the ‘Z-summary’ statistic as a general summary of all the different statistics used. To assess the effect of k-means on Z-summary, we performed the same investigations on both 10 UKBEC brain tissues and on the 13 GTEx brain tissues. Note we focus on brain tissues within GTEx as comparison of preservation only makes sense for tissues that are similar. Within each UKBEC and GTEx tissue GCN, we compared the preservation of all the partitions generated by WGCNA alone with the preservation obtained by applying k-means to each of them. A permutation analysis on 10 tissues generates, for



each tissue t , and for each module p_i within the corresponding tissue network, a vector of 9 Z-summary values corresponding to the preservation of p_i in the other 9 tissues.

Table 2 displays the results of the comparison between WGCNA and k-means. Each table cell indicates the difference between the number of modules preserved after applying k-means, versus the number of modules preserved with standard WGCNA (defined as Z-summary > 10 following the author's recommendation). For example, in subtable (a), FCTX (row) shows 5 more modules preserved in CRBL (column) after applying the k-means method.

From Table 2 it is apparent that there is an overall increase in the number of modules preserved under k-means. In the UKBEC GNCs, there is an improvement in 73 cases (81%), no improvement in 16 cases (17%), and only case with a worse preservation (thalamus in white matter). The average improvement in modules preserved

for UKBEC is 2.1. In the GTEx GCNs, there is an improvement in module preservation in 133 cases (85%), no improvement in only 20 cases (12.8%) and a decreased preservation in just 3 cases. The average number of modules improved by the k-means method is 4.2 (note that in GTEx networks we get higher number of modules per GCN).

This suggests that k-means creates less noisy modules as similarities between tissues are more apparent. Finally, it is worth noting that each tissue is expected to have specific modules, i.e. modules that will be poorly preserved in other tissues because they are exclusive from that tissue, reflecting study-specific or sample-specific gene subsystems.

K-means detects more accurate partitions than WGCNA in simulated data

We wanted to test whether k-means improves the accuracy of partitions P with respect to those obtained

Table 2 Number of new modules from a tissue (rows) that are preserved on another tissue (columns) after applying the k-means to the standard WGCNA partitions

(a) UKBEC brain tissues										
	CRBL	FCTX	HIPP	MEDU	OCTX	PUTM	SNIG	TCTX	THAL	WHMT
CRBL	0	1	2	3	0	5	3	1	3	2
FCTX	5	0	1	5	0	6	3	0	5	3
HIPP	4	2	0	3	0	7	1	3	0	0
MEDU	1	2	4	0	3	2	3	2	1	1
OCTX	7	1	3	6	0	9	6	3	8	6
PUTM	3	1	3	2	1	0	1	1	0	2
SNIG	1	2	1	0	1	1	0	0	0	1
TCTX	3	2	1	3	0	1	4	0	2	4
THAL	2	2	3	1	1	1	0	0	0	-1
WHMT	1	1	0	1	1	0	1	1	0	0

(b) GTEx brain tissues													
	AMYG	ACCT	CAUD	CEHE	CERE	CTEX	FCTX	HIPP	HYPO	NUAC	PUTM	SPIN	SNIG
AMYG	0	0	14	1	2	5	6	3	10	14	7	7	7
ACCT	0	0	0	3	1	3	-1	5	1	0	1	6	2
CAUD	2	3	0	2	-1	5	6	5	6	3	1	2	4
CEHE	2	0	1	0	8	0	0	0	4	1	2	0	0
CERE	1	2	1	15	0	4	1	1	0	0	1	1	1
CTEX	2	4	0	4	3	0	1	2	4	4	4	6	3
FCTX	4	5	6	1	2	4	0	3	0	1	4	1	2
HIPP	0	4	7	2	4	0	8	0	1	2	2	8	0
HYPO	2	9	7	4	5	6	5	11	0	3	4	1	1
NUAC	12	11	7	9	5	9	7	20	9	0	6	4	7
PUTM	1	-3	4	6	1	5	3	8	7	5	0	6	5
SPIN	4	2	1	-1	1	6	2	7	9	5	14	0	4
SNIG	15	12	5	0	0	18	4	9	14	7	13	6	0

under standard WGCNA. To this end we investigated networks based on ‘synthetic data’. The WGCNA package provides a gene expression profiling simulation method `simulateDataExp()`, which is a convenient method for generating artificial data sets that mimic the properties of real datasets. The simulation method works with the eigengene of gene expression for each gene belonging to the module.

The simulation method requires, as arguments: (1) a matrix with the eigengene for each module; (2) the proportion of the total gene pool that one will find within each module; and (3) the number of genes to be simulated. Note that the number of samples we want to simulate appears implicitly as the length of each eigengene (each eigengene has as many components as samples used to construct the GCN). The method returns two elements: (1) a gene expression profiling data set, let us denote it with D , that we can use to construct GCNs; and (2) the ideal clustering partition of the simulated gene expression profiling, here denoted by $P(D)$. Thus, if we rely on the effectiveness of this simulation method, then a simulated data set D , we will prefer a GCN construction algorithm A to algorithm B if the distance between $A(D)$ and $P(D)$ is smaller than between $B(D)$ and $P(D)$, where $A(D)$ and $B(D)$ are the clustering partitions we get after constructing GCNs on D with A and B , respectively. The accuracy of an algorithm A is defined by the similarity of the theoretical optimal partition within the synthetic data to the partition constructed by A .

In order to test whether k-means performs any better than standard WGCNA on simulated data, we constructed a plausible set of simulated gene expression profiles. We used GTEx and test with them both k-means and standard WGCNA on GCN construction.

The accuracy of an algorithm A will be defined as how similar are the theoretical optimal partition within the synthetic data, and the partition constructed by A .

In order to test whether k-means performs any better than standard WGCNA on simulated data, we constructed a plausible set of simulated gene expression profiles. We used GTEx standard WGCNA GCNs (i.e. their eigengenes and module relative size) as the simulation seed for the generation of a synthetic gene expression profile. We focused on the GTEx dataset rather than UKBEC, because the 42 GCNs comprise a usefully heterogeneous network dataset. The simulated data process produced a gene expression profile and a theoretical ideal clustering partition for such profile. We used this theoretical ideal partition to evaluate standard WGCNA and k-means accuracy. To estimate accuracy we use three different statistics: (1) the Rand [25] index, also implemented within WGCNA, the Jaccard coefficient and the similarity index [26], all of them implemented within `clv` R package.

Results for all the experiments appear in Additional file 2. Each row corresponds to a GTEx tissue, the `randsimvswgcna` column corresponds to the Rand index between the ideal partition and that obtained with WGCNA on the simulated data. The `randsimvskm` column corresponds to the same index when using k-means. The other four columns correspond to the Jaccard coefficient and the similarity index.

The k-means refinement generate higher values in all the cases for all three indexes. These results are illustrated in Fig. 4.

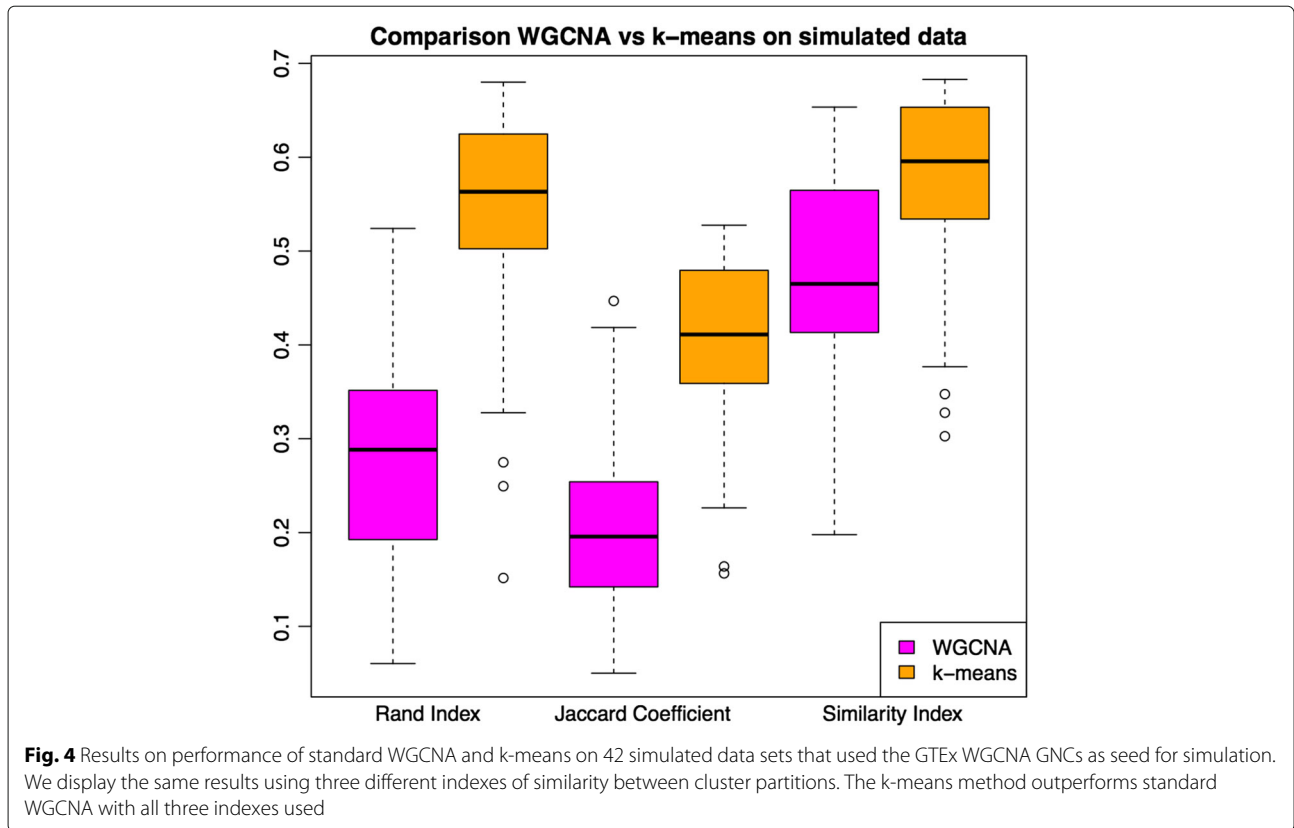
K-means improves functional enrichments

The Gene Ontology [27] is a curated database for gene annotation which can be used for the functional characterization of gene sets. Given a set of genes (i.e. the gene set used to create our GCN), and a subset of those genes (i.e. a module within our partition P), an enrichment analysis can be performed on GO annotations [28] to search for terms in the ontology that are significantly enriched in the subset of genes relative to the full set. The number and strength of significant terms obtained in this way can be used to measure the biological functionality of the module.

Given two different partitions P and P' created from the same TOM, we would prefer the partition that generates more significant GO terms if we assume GO to reflect a biological ground truth as this would suggest that the preferred partition makes more biological sense. We used the `gProfiler` R package [29] to obtain enrichment p -values, avoiding EIA (Electronic Inferred Annotations) terms in GO and requiring a correction for multiple testing with `gSCS`, as developed by the authors of the package. We describe below a series of investigations to characterize the improvement in a module’s biological functionality.

Global annotation term significance

Consider a partition $P = \{p_1, \dots, p_k\}$ of genes arranged into modules p_i , $1 \leq i \leq k$. Now suppose we want to perform a gene set enrichment analysis on each $p_i \in P$ based on the Gene Ontology. GO is a list of ontological terms, organised into three main branches: BP (Biological Process), MF (Molecular Function) and CC (Cellular component). Genes within the database will be associated with a number of terms from each branch. Thus, for each term in GO, and given the list of genes in p_i , we can apply a contingency test, e.g. Fisher exact test [30], under the null hypothesis that the genes in p_i show no significant overlap with the set of genes associated with the term. With an appropriate correction for multiple testing, we define as significant the association of the list of genes in p_i with the corresponding term, when the corrected p -value is < 0.05 . We then aggregate all these p -values for a module in a sin-



gle measure of significance as follows. For each $p_i \in P$, we use

$$s_{GO}(p_i) = \sum_{pvalue_j \in test(p_i, GO)} -\log_{10}(pvalue_j), \quad (4)$$

where $test(p_i, GO)$ is the set of p -values, $pvalue_j$, of significant terms associated with the genes in partition p_i , emerging from the analysis. In this way,

$$S_{GO}(P) = \sum_{p_i \in P} s_{GO}(p_i) \quad (5)$$

can be used to aggregate all the biological signals (i.e. all the significant annotation terms) of a whole partition P . Given a choice of partitions, we prefer P to P' when $S_{GO}(P) > S_{GO}(P')$.

Figure 5a displays for each UKBEC and GTEx GCN, the relative improvement between the standard WGCNA partition, P and the k-means partition, P' by

$$\frac{S_{GO}(P')}{S_{GO}(P)} - 1.$$

The average improvement is 13% (ranging from -22.9% for the GTEx Spleen GCN to 109.1% for the UKBEC Putamen GCN). Overall, there is improvement in all UKBEC tissues and in 34 out of 42 GTEx GCNs, and the overall improvement is significant (paired t-test p -value $2.01e-6$).

Does a higher enrichment implies less informative modules?

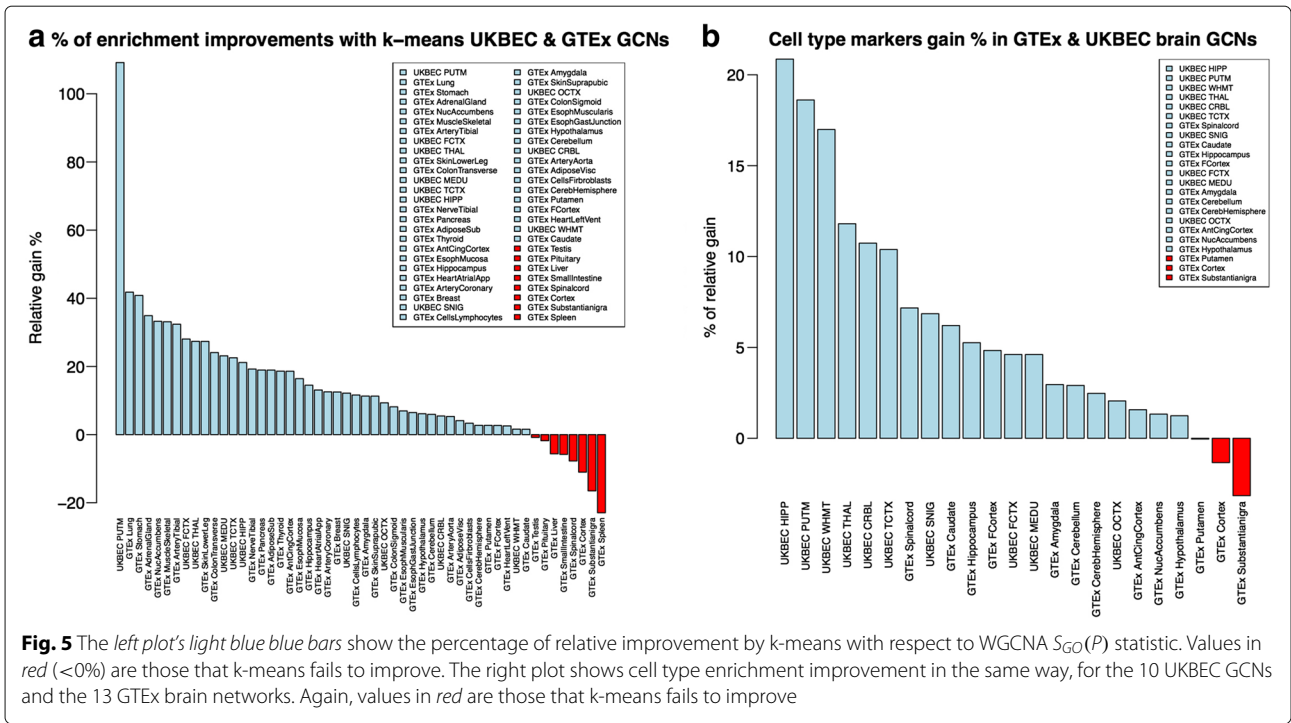
High values of the $S_{GO}(P)$ index are of interest, as we prefer a partition P over P' if $S_{GO}(P) > S_{GO}(P')$. However, it is possible that modules show better S_{GO} values after k-means because the module have more annotation terms that are generic, and therefore less descriptive about the specifics of the tissue studied. In order to assess this, we applied the notion of information content [31]. We used the GOSim package [32] which applies information-based metrics to Gene Ontology terms. The metric $IC(t)$ for a term t belonging to an ontology is defined as:

$$IC(t) = -\log P(t), \quad (6)$$

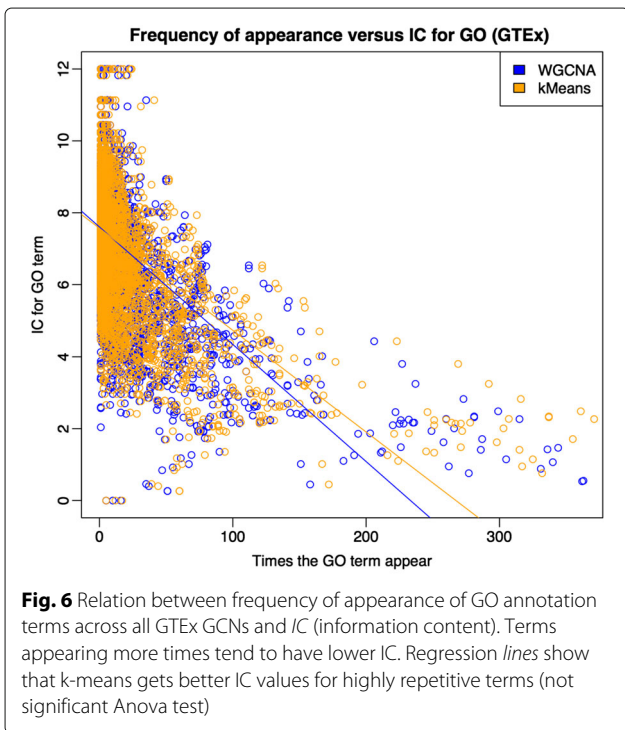
where $P(t)$ is the probability of observing t within the annotations available within that ontology.

Ideally, we prefer modules with more GO terms, which are more significant (i.e. more reliably defining the network module) and more informative (i.e. terms that are highly specific for the sample's tissue). From previous sections we know we have more significant networks thanks to S_{GO} . But is k-means capable of not only improving significance but also of maintaining the level of information of the modules if not increasing it?

Figure 6 displays the differences between standard WGCNA and k-means in the number of times a term appears across all 44 GTEx networks (x-axis) versus their



IC values (y-axis). Each point represents a significant GO term, obtained by gProfileR as described above. We may expect that terms with lower IC values appear more frequently within the GCNs' functional characterization because they are more abundant on the Gene Ontology.



The plot shows that both for kMeans and standard WGCNA there is a clear tendency for the more frequent terms to be also those with lower information content (Pearson correlation -0.58 , p -value $< 2.2e - 16$).

Is the overall IC obtained by k-means degraded as a consequence of obtaining more significant terms per GCN in comparison with standard WGCNA? To assess this we regressed the information content of the significant annotation terms against the frequency of appearance in the GCN annotation sets. We found a tendency towards higher IC in k-means GCNs. This suggests that k-means annotations are more specific, and therefore more useful.

Is the increase in enrichment better than random?

In "Dynamics of k-means when working on 1-TOM distance space" section we noted that one of the changes within cluster partitions after applying k-means is that module sizes change and many modules will increase their size considerably. It is fair to assume that modules increasing their size in genes, will also increase their s_{GO} enrichment. There is a significant Pearson correlation between increase in module size after k-means and increase in number of significant annotation terms ($r = 0.42$, p -value $2.2e-16$). The question arises, therefore, of what is the real contribution of k-means in comparison to a random shift of genes between modules?

In order to answer this question, for each of the 42 GTEX tissues and their corresponding WGCNA and k-means

partitions, we identified those genes that were changed at the WGCNA partition to create the k-means one. Then, in a single step, we randomly assigned these genes to other modules in such a way that we kept the same module sizes obtained with k-means. Via this algorithm, we produced new partitions in which the genes that remained unchanged from WGCNA to k-means stayed at the same modules, but those genes that were changed by k-means were again changed but this time in a random fashion.

Figure 7 shows the results of this investigation. Plot (a) shows, for all modules of all GCNs, the $S_{GO}(P)$ statistic. Plot (b) shows the number of significant GO terms.

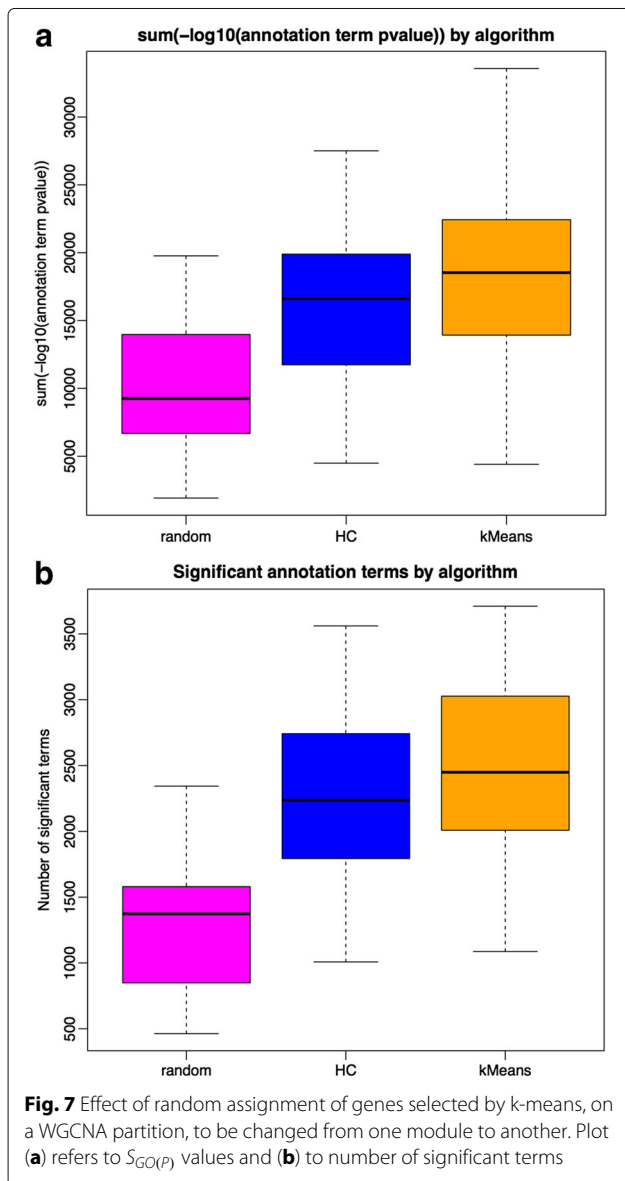
In 89% of the modules, k-means finds the same number (18%) or more (70%) significant GO terms than the random placement of misplaced genes (paired t-test

p -value $< 2.2e - 16$). 88% of the final modules show equal (15%) or better (73%) $S_{GO}(P)$ index using k-means compared to random (paired t-test p -value $< 2.2e - 16$). Aggregating the results by tissue, k-means placement performs better in all the cases. Interestingly, the random placement of misplaced genes prevents enrichment at the WGCNA partition it starts with (i.e. comparing the magenta with the blue plots). This is important because even though many genes at the WGCNA partition are not touched by the random approach, moving genes randomly will nevertheless worsen these genes' functional annotations. This suggests that both the number of significant terms and the $S_{GO}(P)$ index have a reasonable sensitivity.

K-means improves brain specific cell type marker enrichment

One interesting property of WGCNA GCNs is that partitions created from them can be useful when studying cell-type specific gene networks. In studies where samples come from bulk tissue, it is most likely that these samples will be comprised of different cell material. In consequence, the gene expression profiles obtained from them should reflect this heterogeneity in some way. WGCNA's GCNs handle this heterogeneity in an elegant and convenient manner: they often generate gene clusters within partitions which are specialized on a given cell type, i.e. they present a highly significant enrichment of markers (i.e. genes which are differentially expressed) for a given cell type [2, 8, 13, 33].

We wanted to assess the effects of k-means on this particular feature. To do this, we used three different resources defining cell-type specific gene sets. These were WGCNA's brain lists, [34], and two alternative brain specific sources, labelled here *External* [35] and *Cahoy* [36]. We evaluated each partition's modules from the 10 UKBEC GCNs and the 13 brain tissue GTEx GCNs, using both standard WGCNA and k-means. This evaluation generated two matrices of p -values (i.e. one for WGCNA and one for k-means), with each gene dataset in a row and each specific module from each of the 10 networks in a column. P -values reflect a Fisher's exact test for whether there is significant concentration of the corresponding gene sets in the tested module. We include in Additional files 1 and 2, the results for standard WGCNA and k-means, on the 10 UKBEC GCNs. Note that, in these plots, both columns and rows have been clustered based on $-\log_{10}(p$ -values) so it can be better seen how modules from different tissues cluster together at columns, and also how different gene sets cluster among rows. These heat-maps reveal strongly clustered areas corresponding to groups of cell-type specific genes sets within most, if not all, of the tissues. More specifically, we see four groups cell-type specific gene sets corresponding to microglia,



astrocytes, oligodendrocytes and neurons (in order from top right to bottom left).

In the UKBEC k-means heat-map (Additional file 3), using a significance cut-off of 10^{-4} (to account for multiple testing), almost 65% of the modules show cell type enrichment (i.e. 91 modules in total). Within these, 86 modules show a single cell type signal. In the WGCNA heat-map (Additional file 4), 63% of the modules show cell type enrichment (87 modules in total), with 85 showing single cell type signals.

Figure 5b compares the two enrichment matrices, by aggregating all the cell-type enrichment $-\log_{10}$ transformed p -values as we did for the Gene Ontology enrichment in “K-means improves functional enrichments” section. Each bar represents the sum of all values of the corresponding heat-map, for modules of the given tissue. According to this, we always see an improvement in UKBEC networks and in most of the the GTEx networks. The overall improvement is significant (paired t-test p -value 0.000193).

Conclusions

Our study shows that an additional k-means step, when used as an adjunct to WGCNA, improves the partitions generated from gene co-expression networks. Our method is not an alternative to WGCNA, instead it is an additional step to the standard WGCNA pipeline. Indeed, our method can be applied to any general hierarchical clustering algorithm, and as such it could be usefully applied to any hierarchical clustering based approach for network generation, not just gene co-expression networks.

We evaluated our method using two contrasting gene expression datasets representing a variety of different tissues, one obtained with microarray technology (the UKBEC dataset on 10 brain tissues), and the other with RNA-seq (the GTEx dataset on 42 tissues, which includes 13 brain tissues). Using a variety of approaches, we demonstrate improved performance of our k-means method in both datasets. Furthermore, we also demonstrate improved performance using simulated data generated from the GTEx dataset.

We show via these analyses that it is possible to obtain better partitions for the same networks via our k-means method. Our method generates modules with fewer misplaced genes with respect to their eigengene, and this implies that the eigengene is a better representative of the phenomena hidden behind the particular set of genes belonging to the module.

Using Gene Ontology enrichment analyses, we also show that our partitions are enriched for biological functionality. Statistically significant $S_{GO}(P)$ enrichment is seen in all 10 UKBEC CGNs and in 34 out of the 42 GTEx GCNs.

Our partitions have improved modules preservation, which also suggests that the clustering is more accurate from a biological point of view. Although some gene modules are specific of each tissue (and therefore show poor preservation in other tissues), it is a desirable property of most GCN partitions to be highly replicable under the assumption that a preserved module is more likely to be a genuine module. Our analyses suggest that k-means favours the creation of more genuine modules and these results are seen in both UKBEC and GTEx GCNs.

Our k-means method also creates partitions in which gene sets representative of specific brain cell types are seen in modules with increased statistical significance. This suggests, once again, more biologically genuine modules.

GCN construction is likely to become an increasingly important analysis, as genomics and transcriptomics are increasingly applied to aid clinical diagnosis and prognosis. Methods that generate more reliable and robust gene GCNs will enable improved prediction of inter-gene relationships and gene function, with a variety of applications.

Availability and requirements

UKBEC data [37] has accession code GSE46706. All information about tissues, samples and quality control can be found there. GTEx RPKM gene expression V6 was used in this paper and downloaded from the GTEx portal: <http://gtexportal.org/home>. Regarding the software we present here, this is the availability and requirements.

Project name: km2gcn

Project homepage: <https://github.com/juanbot/km2gcn>

Operating system: Linux/Windows/Mac

Programming language: R

Other requirements: WGCNA R package and gProfileR

License: LGPL

Additional files

Additional file 1: Lists tissues, samples and genes used for the creation of each GCN. (CSV 4 kb)

Additional file 2: Includes comparative results of standard WGCNA and k-means on simulated data. (CSV 4 kb)

Additional file 3: The second one corresponds to k-means. Both on UKBEC datasets. Values higher than 20 are set to 20. Colors at the top of columns correspond to tissues, the tissue legend is at the bottom of columns and cell marker gene set used on the right side. (JPG 1177 kb)

Additional file 4: Heat-maps showing $-\log_{10}(p\text{-values})$ from Fisher's Exact test on significant concentration of specific cell marker gene sets (rows) on each tissue module (columns). The one within the Additional file 1 corresponds to the standard WGCNA. (JPG 1198 kb)

Abbreviations

BP: Biological process category of the gene ontology; CC: Cellular component category of the gene ontology; EIA: Electronic inferred annotations; GBA: Guilty by association; GCN: Gene co-expression network; GO: Gene ontology; GTEx: Genotype-tissue expression project; HC: Hierarchical clustering; MF: Molecular function category of the gene ontology; MM: Module membership

of a gene; SFT: Scale free topology; TOM: Topological overlap matrix; UKBEC: United Kingdom brain expression consortium; WGCNA: Weighted gene-co expression network

Acknowledgements

We acknowledge support from the National Institute for Health Research (NIHR) Biomedical Research Centre based at Guy's and St Thomas' and the NIHR Biomedical Research Centre at South London and Maudsley, National Health Service (NHS) Foundation Trust and King's College London. The views expressed are those of the authors and not necessarily those of the NHS, the NIHR or the Department of Health.

United Kingdom Brain Expression Consortium

What follows is the list of the rest of members of the UKBEC consortium. Adaikalavan Ramasamy

- Department of Medical & Molecular Genetics, King's College London, Guy's Hospital, London, UK
- Reta Lila Weston Research Laboratories, Department of Molecular Neuroscience, University College London (UCL) Institute of Neurology, London, UK
- Jenner Institute, University of Oxford, Oxford, UK

Daniah Trabzuni

- Reta Lila Weston Research Laboratories, Department of Molecular Neuroscience, University College London
- Department of Genetics, King Faisal Specialist Hospital and Research Centre, Riyadh, Saudi Arabia

Colin Smith

- Department of Neuropathology, MRC Sudden Death Brain Bank Project, University of Edinburgh, Edinburgh, UK

Robert Walker

- Department of Neuropathology, MRC Sudden Death Brain Bank Project, University of Edinburgh, Edinburgh, UK

Funding

This work was supported by the UK Medical Research Council (MRC) through Project Grant (G511492 to J.H., M.E.W. and M.R.). S.G. was supported by the Alzheimer's Research UK through a PhD Fellowship.

Authors' contributions

JB, JV and PF developed the improvement to WGCNA. JB implemented the software package and JV, PF, SG and KD contributed to software evaluation. JB with particular contributions from all authors wrote the manuscript. JH, CL, MR and mainly MW contributed to study design and reviewed the manuscript. All authors read, reviewed, and approved the final manuscript.

Competing interests

Author MW is an employee of Genomics plc, a company providing genomic analysis services to the pharmaceutical and health care sectors. His involvement in the conduct of this research was solely in his capacity as a Reader in Statistical Genetics at King's College London. The rest of authors declare that they have no competing interests.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Department of Molecular Neuroscience, Institute of Neurology, University College London, Queen Square, WC1N 3BG London, UK. ²Department of Medical & Molecular Genetics, School of Medical Sciences, King's College London, Guy's

Hospital, SE1 9RT London, UK. ³Istituto di Ricerca Genetica e Biomedica, CNR, Cittadella Universitaria di Monserrato, 09042 Monserrato, CA, Italy.

Received: 26 August 2016 Accepted: 17 March 2017

Published online: 12 April 2017

References

1. Carpenter AE, Sabatini DM. Systematic genome-wide screens of gene function. *Nat Rev Genet.* 2004;5(1):11–22. doi:10.1038/nrg1248.
2. Parikshak NN, Gandal MJ, Geschwind DH. Systems biology and gene networks in neurodevelopmental and neurodegenerative disorders. *Nat Rev Genet.* 2015;16(8):441–58.
3. Mostafavi S, Morris Q. Combining many interaction networks to predict gene function and analyze gene lists. *Proteomics.* 2012;12(10):1687–1696. doi:10.1002/pmic.201100607. Accessed 07 Sept 2015.
4. Gamazon ER, Wheeler HE, Shah KP, Mozaffari SV, Aquino-Michaels K, Carroll RJ, Eyster AE, Denny JC, Nicolae DL, Cox NJ, Im HK. A gene-based association method for mapping traits using reference transcriptome data. *Nat Genet.* 2015;47(9):1091–8. doi:10.1038/ng.3367. Accessed 11 Sept 2015.
5. Langfelder P, Luo R, Oldham MC, Horvath S. Is my network module preserved and reproducible? *PLoS Comput Biol.* 2011;7(1):1001057. doi:10.1371/journal.pcbi.1001057. Accessed 09 Sept 2015.
6. Langfelder P, Horvath S. WGCNA: an R package for weighted correlation network analysis. *BMC Bioinforma.* 2008;9(1):559. doi:10.1186/1471-2105-9-559. Accessed 07 Sept 2015.
7. Bettencourt C, Ryten M, Forabosco P, Schorge S, Hershenson J, Hardy J, Houlden H. Insights from Cerebellar Transcriptomic analysis into the Pathogenesis of Ataxia. *JAMA Neurol.* 2014;71(7):831. doi:10.1001/jamaneurol.2014.756. Accessed 15 Sept 2015.
8. Forabosco P, Ramasamy A, Trabzuni D, Walker R, Smith C, Bras J, Levine AP, Hardy J, Pocock JM, Guerreiro R, Weale ME, Ryten M. Insights into TREM2 biology by network analysis of human brain gene expression data. *Neurobiol Aging.* 2013;34(12):2699–714. doi:10.1016/j.neurobiolaging.2013.05.001. Accessed 15 Sept 2015.
9. Mencacci NE, Rubio-Agusti I, Zdebek A, Asmus F, Ludtmann MR, Ryten M, Plagnol V, Hauser AK, Bandres-Ciga S, Bettencourt C, Forabosco P, Hughes D, Soutar MP, Peall K, Morris H, Trabzuni D, Tekman M, Stanescu H, Kleta R, Carecchio M, Zorzi G, Nardocci N, Garavaglia B, Lohmann E, Weissbach A, Klein C, Hardy J, Pittman A, Foltynie T, Abramov A, Gasser T, Bhatia K, Wood N. A Missense mutation in KCTD17 causes Autosomal Dominant Myoclonus-Dystonia. *Am J Hum Genet.* 2015;96(6):938–47. doi:10.1016/j.ajhg.2015.04.008. Accessed 15 Sept 2015.
10. Allen JD, Xie Y, Chen M, Girard L, Xiao G. Comparing statistical methods for constructing large scale gene networks. *PLoS ONE.* 2012;7(1):29348. doi:10.1371/journal.pone.0029348. Accessed 08 Sept 2015.
11. Ballouz S, Verleyen W, Gillis J. Guidance for RNA-seq co-expression network construction and analysis: safety in numbers. *Bioinformatics.* 2015;31(13):2123–30. doi:10.1093/bioinformatics/btv118.
12. Filosi M, Visintainer R, Riccadonna S, Jurman G, Furlanello C. Stability indicators in network reconstruction. *PLoS ONE.* 2014;9(2):89815. doi:10.1371/journal.pone.0089815. Accessed 08 Sept 2015.
13. Miller JA, Woltjer RL, Goodenbour JM, Horvath S, Geschwind DH. Genes and pathways underlying regional and cell type changes in Alzheimer's disease. *Genome Med.* 2013;5(5):48. doi:10.1186/gm452. Accessed 04 Feb 2016.
14. Hartigan JA, Wong MA. Algorithm AS 136: a k-means clustering algorithm. *Appl Stat.* 1979;28(1):100. doi:10.2307/2346830. Accessed 07 Sept 2015.
15. Langfelder P, Zhang B, Horvath S. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for R. *Bioinformatics.* 2008;24(5):719–20. doi:10.1093/bioinformatics/btm563. Accessed 29 Jan 2016.
16. Arthur D, Vassilvitskii S. k-means++: the advantages of careful seeding. In: *SODA '07 Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.* New Orleans: ACM (Association for Computing Machinery); 2007.
17. Albert R. Scale-free networks in cell biology. *J Cell Sci.* 2005;118(21):4947–57. doi:10.1242/jcs.02714. Accessed 09 Sept 2015.
18. Chuang YF, Tanaka T, Beason-Held LL, An Y, Terracciano A, Sutin AR, Kraut M, Singleton AB, Resnick SM, Thambisetty M. FTO genotype and

- aging: pleiotropic longitudinal effects on adiposity, brain function, impulsivity and diet. *Mol Psychiatry*. 2015;20(1):140–7. doi:10.1038/mp.2014.49. Accessed 03 Feb 2016.
19. Stampfel G, Kazmar T, Frank O, Wienerroither S, Reiter F, Stark A. Transcriptional regulators form diverse groups with context-dependent regulatory functions. *Nature*. 2015. doi:10.1038/nature15545. Accessed 03 Feb 2016.
 20. Cookson MR. LRRK2 pathways leading to Neurodegeneration. *Curr Neurol Neurosci Rep*. 2015;15(7). doi:10.1007/s11910-015-0564-y. Accessed 03 Feb 2016.
 21. The GTEx Consortium, Ardlie KG, Deluca DS, Segre AV, Sullivan TJ, Young TR, Gelfand ET, Trowbridge CA, Maller JB, Tukiainen T, Lek M, Ward LD, Kheradpour P, Iriarte B, Meng Y, Palmer CD, Esko T, Winckler W, Hirschhorn JN, Kellis M, MacArthur DG, Getz G, Shabalin AA, Li G, Zhou YH, Nobel AB, Rusyn I, Wright FA, Lappalainen T, Ferreira PG, Ongen H, Rivas MA, Battle A, Mostafavi S, Monlong J, Sammeth M, Mele M, Reverter F, Goldmann JM, Koller D, Guigo R, McCarthy MI, Dermitzakis ET, Gamazon ER, Im HK, Konkashbaev A, Nicolae DL, Cox NJ, Flutre T, Wen X, Stephens M, Pritchard JK, Tu Z, Zhang B, Huang T, Long Q, Lin L, Yang J, Zhu J, Liu J, Brown A, Mestichelli B, Tidwell D, Lo E, Salvatore M, Shad S, Thomas JA, Lonsdale JT, Moser MT, Gillard BM, Karasik E, Ramsey K, Choi C, Foster BA, Syron J, Fleming J, Magazine H, Hasz R, Walters GD, Bridge JP, Miklos M, Sullivan S, Barker LK, Traino HM, Mosavel M, Siminoff LA, Valley DR, Rohrer DC, Jewell SD, Branton PA, Sobin LH, Barcus M, Qi L, McLean J, Hariharan P, Um KS, Wu S, Tabor D, Shive C, Smith AM, Buia SA, Undale AH, Robinson KL, Roche N, Valentino KM, Britton A, Burges R, Bradbury D, Hambright KW, Seleski J, Korzeniewski GE, Erickson K, Marcus Y, Tejada J, Taherian M, Lu C, Basile M, Mash DC, Volpi S, Struewing JP, Temple GF, Boyer J, Colantuoni D, Little R, Koester S, Carithers LJ, Moore HM, Guan P, Compton C, Sawyer SJ, Demchok JP, Vaught JB, Rabiner CA, Lockhart NC, Ardlie KG, Getz G, Wright FA, Kellis M, Volpi S, Dermitzakis ET. The genotype-tissue expression (GTEx) pilot analysis: multitissue gene regulation in humans. *Science*. 2015;348(6235):648–60. doi:10.1126/science.1262110. Accessed 07 Sept 2015.
 22. Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, Szczesniak MW, Gaffney DJ, Elo LL, Zhang X, Mortazavi A. A survey of best practices for RNA-seq data analysis. *Genome Biol*. 2016;17(1). doi:10.1186/s13059-016-0881-8. Accessed 11 Dec 2016.
 23. Leek JT, Johnson WE, Parker HS, Fertig EJ, Jaffe AE, Storey JD. sva: Surrogate Variable Analysis. R package version 3.22.0. 2016. <https://bioconductor.org/packages/release/bioc/manuals/sva/man/sva.pdf>.
 24. Vapnik VN. The nature of statistical learning theory. New York: Springer; 2000.
 25. Rand WM. Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc*. 1971;66(336):846. doi:10.2307/2284239. Accessed 30 Nov 2016.
 26. Lange T, Roth V, Braun ML, Buhmann JM. Stability-based validation of clustering solutions. *Neural Comput*. 2004;16(6):1299–323. doi:10.1162/089976604773717621. Accessed 08 Jan 2017.
 27. The Gene Ontology Consortium. Gene ontology consortium: going forward. *Nucleic Acids Res*. 2015;43(D1):1049–56. doi:10.1093/nar/gku1179. Accessed 18 Jan 2016.
 28. Khatri P, Draghici S. Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*. 2005;21(18):3587–95. doi:10.1093/bioinformatics/bti565. Accessed 03 Feb 2016.
 29. Reimand J, Kull M, Peterson H, Hansen J, Vilo J. g:Profiler—a web-based toolset for functional profiling of gene lists from large-scale experiments. *Nucleic Acids Res*. 2007;35(Web Server):193–200. doi:10.1093/nar/gkm226. Accessed 18 Jan 2016.
 30. Rodel E, Fisher RA. *Statistical Methods for Research Workers*, 14. Aufl., Oliver & Boyd, Edinburgh, London 1970. XIII, 362 S., 12 Abb., 74 Tab., 40 s. *Biom Z*. 1971;13(6):429–30. doi:10.1002/bimj.19710130623. Accessed 03 Feb 2016.
 31. Shannon CE. A mathematical theory of communication. *Bell Syst Tech J*. 1948;27(3):379–423. doi:10.1002/j.1538-7305.1948.tb01338.x. Accessed 02 Dec 2016.
 32. Fröhlich H, Speer N, Poustka A, Reißbarth T. GOSim – an R-package for computation of information theoretic GO similarities between terms and gene products. *BMC Bioinforma*. 2007;8(1):166. doi:10.1186/1471-2105-8-166. Accessed 02 Dec 2016.
 33. Miller JA, Horvath S, Geschwind DH. Divergence of human and mouse brain transcriptome highlights Alzheimer disease pathways. *Proc Natl Acad Sci*. 2010;107(28):12698–703. doi:10.1073/pnas.0914257107. Accessed 04 Sept 2015.
 34. Miller JA, Cai C, Langfelder P, Geschwind DH, Kurian SM, Salomon DR, Horvath S. Strategies for aggregating gene expression data: the collapseRows R function. *BMC Bioinforma*. 2011;12(1):322. doi:10.1186/1471-2105-12-322. Accessed 07 Feb 2016.
 35. Zeisel A, Munoz-Manchado AB, Codeluppi S, Lonnerberg P, La Manno G, Jureus A, Marques S, Munguba H, He L, Betsholtz C, Rolny C, Castelo-Branco G, Hjerling-Leffler J, Linnarsson S. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science*. 2015;347(6226):1138–42. doi:10.1126/science.aaa1934. Accessed 25 May 2016.
 36. Cahoy JD, Emery B, Kaushal A, Foo LC, Zamanian JL, Christopherson KS, Xing Y, Lubischer JL, Krieg PA, Krupenko SA, Thompson WJ, Bares BA. A Transcriptome database for Astrocytes, Neurons, and Oligodendrocytes: a new resource for understanding brain development and function. *J Neurosci*. 2008;28(1):264–78. doi:10.1523/JNEUROSCI.4178-07.2008. Accessed 04 Sept 2015.
 37. Ramasamy A, Trabzuni D, Forabosco P, Smith C, Walker R, Dillman A, Sveinbjornsdottir S, North American Brain Expression Consortium (NABEC), UK Brain Expression Consortium (UKBEC), Hardy J, Weale ME, Ryten M. Genetic evidence for a pathogenic role for the vitamin D3 metabolizing enzyme CYP24a1 in multiple sclerosis. *Mult Scler Relat Disord*. 2014;3(2):211–9. doi:10.1016/j.msard.2013.08.009.

Submit your next manuscript to BioMed Central and we will help you at every step:

- We accept pre-submission inquiries
- Our selector tool helps you to find the most relevant journal
- We provide round the clock customer support
- Convenient online submission
- Thorough peer review
- Inclusion in PubMed and all major indexing services
- Maximum visibility for your research

Submit your manuscript at
www.biomedcentral.com/submit

