Supplementary Methods

Gene expression datasets of breast invasive carcinoma were downloaded from The Cancer Genome Atlas (TCGA) data portal and from Boersma *et al*.(Boersma, et al. (2008)) and Esserman *et al* (Esserman, et al., 2012). Breast cancer samples were classified into four clinical stages by the study authors using the latest AJCC (American Joint Committee on Cancer) staging system (Edge, 2010). In this study, we only considered protein-coding genes annotated in the NCBI RefSeq database. The starting data for all downstream analyses were quantile-normalized log2 expression ratios. To compute p-values of differential gene expression, SAM (Tusher, et al., 2001) was used.

The list of mutated genes was downloaded from the Catalogue of Somatic Mutations In Cancer (COSMIC) (12 June 2012) (Pleasance, et al., 2010).

List of signaling domains and non-signaling domains was downloaded from the SMART database (Letunic, et al., 2012).

List of manually annotated protein complexes in human was downloaded from the CORUM database (Ruepp, et al., 2010).

Construction of co-expression networks from gene expression data

To discover M-modules in multiple networks, we first represent each gene expression dataset as a graph or network by regarding each gene as a node and expression correlation between two genes as an edge. There were 8772 genes that are present in all four TCGA datasets representing four clinical stages of breast cancer. We used these genes to construct a co-expression network for each clinical stage. For a gene pair, we quantified their co-expression using Pearson correlation coefficient. To avoid indirect correlation, we used first-order partial Pearson correlation coefficient (Watson-Haigh, et

al., 2010). This correlation coefficient was used as the edge weight in the gene co-expression network. We extracted the common largest connected components in the original co-expression networks for downstream analyses. There were 7,737 genes that are shared in all four networks.

***M-module* Parameter optimization (related to supplemental Figure 1)**

There are two parameters in the *M-module* algorithm. The parameter α controls the relative contributions of prior knowledge and topological feature to seed selection. The parameter β controls the number of seeds. To determine the optimal α value, we ran *M-module* using different values of alpha (0.1, 0.3, 0.5, 0.7, 0.9). We then used the sets of discovered modules as features to test the performance of breast-cancer-stage prediction, just like what we did with alpha=0.5 in previous submission. We chose α=0.5 because this value gave the best balanced performance (Supplementary Figure 1C,D). To choose the optimal number of seeds, we performed module search starting with 1% of the nodes in the networks as seeds. We kept increasing the number of seeds until the number of significant M-modules did not increase (Supplementary Figure 1B). This percentage is selected as the optimal value. We ended up using top 5% as the threshold.

Benchmarking *M-module* algorithm using simulated networks (related to Figure 2A)

M-module was benchmarked against the following existing algorithms: *JointCluster (Narayanan, et al., 2010)*, *Tensor Clustering* (Li, et al., 2011), *Consensus Clustering* (Lancichinetti and Fortunato, 2012), and *Spectral Clustering* (Newman, 2006). Among these methods, *JointCluster* and *Tensor Clustering* are specifically designed for handling multiple gene networks whereas *Consensus Clustering* and *Spectral Clustering* are general clustering algorithms.

The simulated networks were generated following the strategy by Narayanan *et al*.

Each simulated network contains 256 nodes and 2048 edges such that every node has 16 edges. Each network contains 8 true clusters of equal number of nodes (32). Each node is randomly assigned to a cluster. For a given node, we randomly connect the 16 edges incident on the node based on the parameter k_{out}, which specifies the expected fraction of edges that connect a node in a cluster to nodes outside the cluster. Specifically, a random number between 0 and 1 is first generated. An edge is randomly connected to a node within the cluster if the random number is smaller than k_{out}. Otherwise, the edge is connected to a node outside the cluster. This process is repeated 16 times for each node. The larger $k_{out}$is, the more edges are connected from a node to nodes outside a cluster and thus the noisy a cluster is. We varied k_{out} from 0.1 to 0.6 with a step size of 0.1 to construct networks with different levels of noise for clusters.

Identification of significantly changed interactions between two adjacent component modules of an M-module (related to Figure 3A)

Given an edge, we compute its weight change between two adjacent co-expression networks as the absolute value of the difference in its weights in the two networks. To obtain the significance of the weight change, we randomize the two adjacent co-expression networks separately by degree-preserved edge shuffling. Each network is randomized 10,000 times. We used the randomized networks to construct the null distribution for weight change. The empirical p-value for a given edge weight change is calculated using the null distribution. The significance level is set at 0.05.

Correlation between module activity dynamics and connectivity dynamics (related

to Figure 3C)

We define module activity dynamics and connectivity dynamics for two adjacent cancer stages. For a given module, its activity dynamic score is defined as the difference in its mean activities across patient samples of the two stages. Its connectivity dynamic score is the same as described in the Online Methods section, which is the l_2 norm of the matrix subtraction of the two adjacent matrices normalized by the number of genes in the module. We computed the Pearson correlation between module activity dynamic scores and connectivity dynamic scores.

Protein domain analysis of 4-module genes (related to Figure 3F)

The occurrence frequency of a protein domain *i* in genes of dynamic 4-modules is defined as $f_i^{dyn} = n_i^{dyn}/n_i^{all}$, where n_i^{dyc} and n_i^{all} are the numbers of occurrences of protein domain *i* in genes of dynamic 4-modules and all genes in the network, respectively. The occurrence frequency of a protein domain *i* in genes of static 4-modules is defined similarly as $f_i^{stat} = n_i^{stat}/n_i^{all}$. We next define an occurrence frequency difference as $d_i = f_i^{dyn} - f_i^{stat}$. Notice that a d_i value of +1 indicates that domain *i* is unique to genes of dynamic modules and a d_i value of -1 indicates that domain i is unique to genes of static modules. To compute the p-value of the domain occurrence enrichment in either sets of genes, we used the binomial distribution in which the probability of domain *i* with a positive d_i value is 0.5, the number of trial is the number of protein domains tested and the number of success is the number of protein domains with a positive d_i value.

Predicting cancer stages using multi-class SVM classifier (related to Figure 4)

Given training data $x^{[i]} \in R^n, i = 1, \ldots, l$ in two classes, and an indicator vector $y \in R^l$

such that $y_i \in \{1, -1\}$, the binary SVM solves the following optimization problem (Boser, et al., 1992).

$$min_{w,g,\xi} w^T w + C \sum_{i=1}^{l} \xi_i$$

$$s.t. \quad y_i(w^T \phi(x_i) + b \geq 1 - \xi_i$$

$$\xi_i \geq 0, i = 1, \ldots, l$$

where $\phi(\boldsymbol{x_i})$ is a kernel function mapping $\boldsymbol{x_i}$ into a higher-dimensional space, $\boldsymbol{w}$ is the normal vector to the hyperplane corresponding to the classifier, b is the intercept of the hyperplane for the classifier, ξ_i is the relaxation parameter for smoothness of the SVM's response and $C > 0$ is the regularization parameter. The multi-class classification $y_i \in \{1, \ldots, k\}$ is implemented via multiple binary SVM classifiers {Hsu, 2002 #1335}. Briefly, the one-against-one approach for multi-class classification casts it as constructing multiple binary classification problems. If k is the number of classes, then $k(k-1)/2$ subclassifiers are trained, each of which on data from two classes. To train a subclassifier for the *ith* and *jth* classes, the binary classification problem is defined as:

$$min_{w_{ij}, b_{ij}, \xi_{ij}} \frac{1}{2} \left(w_{ij}\right)^T w_{ij} + C \sum_t \xi_{ij}^t$$

subject to $y_t \left(\left(w_{ij}\right)^T \phi(x_t) + b_{ij} \right) \geq 1 - \xi_{ij}^t \; and \; \xi_{ij}^t \geq 0, t = 1, \ldots, l_{ij}$, where w_{ij} for the normal vector for the hyperplane corresponding to the classifier between the *ith* and *jth* classes, and l_{ij} is the number of samples in the *ith* and *jth* classes.

Cross validation using unbalanced datasets (related to Supplemental Figures 3)

To measure unbiased classification performance, we used 5-fold cross validation. Patient samples in each stage were divided into 5 subsets of equal size. At each iteration, four subsets were used for training a classifier and the remaining subset for testing.

Since the datasets are unbalanced (i.e. different number of patients at each clinical stage), we adopted two commonly used strategies to transform the imbalanced data to balanced data (He and Garcia, 2009): randomly over-sampling and synthetic sampling. These two procedures are briefly described below.

Given a data set $S = \{\boldsymbol{x}_i, y_j\}, i = 1, \ldots, m$ with m samples, where $\boldsymbol{x_i}$ is an instance, and $y_i \in \{1,2, \ldots, C\}$ is class label associated with instance $\boldsymbol{x_i}$. C=2 means that it is a binary classification problem. Let $S_{min} \subset S, S_{maj} \subset S$ be the set of minority and majority class examples. The random oversampling strategy randomly samples $|S_{maj}| - |S_{min}|$ instances in the minority class, where $|S_{min}|$ and $|S_{maj}|$ are the numbers of samples in each class. An new instance $\boldsymbol{x}^* = (x_1^*, x_x^*, \ldots, x_k^*)$ is constructed, where $x_l^* (1 \leq l \leq k)$ is the value randomly sampled from the *lth* values of all the instance within the minority class. The synthetic sampling strategy is based on data generation. To create a synthetic instance, randomly select one of the K-nearest neighbors, and construct a new instance as

$$x^* = x_i + (\bar{x} - x_i) \times \gamma$$

where x_i is an instance in the minority class, $\bar{x}$ is one of the K-nearest neighbors, and $\gamma \in [0,1]$ is a random number.

Two measures were employed to evaluate the performance: accuracy and ROC curve. Accuracy is defined as the number of patient samples classified correctly divided by the total number of samples classified. Please see below section for the computation of ROC curve.

We trained and tested classifiers on balanced datasets created using the above two strategies. We found that the overall performance rankings of the different feature sets were the same when using the two different data transformation strategies although the

absolute values of the performance metrics differ. Therefore, we chose the randomly oversampling strategy that gave better classification performance for all analyses in this study.

Receiver operating characteristic curve for multi-class classification (related to Figure 2, 4, and Supplemental Figure 5, 6, 9)

ROC curve is used to evaluate balanced performance. It depicts how the sensitivity (SN) and specificity (SP) change at various parameter settings. Sensitivity is defined as the proportion of the true positives that are predicted as such. Specificity is defined as the proportion of true negatives that are predicted as such. In this paper, both the work on simulated networks and the work on disease-stage classification are multi-class classification problem. In the case of simulated networks, a node can be classified as belonging to one of 8 clusters. In the case of real networks, a sample can be classified as being one of 4 stages. To this end, we adopted the strategy by (Fawcett, *et al* 2006) , which handles k classes by producing k different ROC curves, one for each class. In detail, for class i, the ROC curve i plots the classification performance using class i as the positive class and all other classes as the negative classes. The overall ROC curve for the multi-class classification is the average of the k ROC curves.

Analysis of factors affecting the classification performance on the TCGA dataset (related to Supplemental Figures 4 and 5)

We tested different cross validation schemes, including 4-fold and 8-fold cross validations (Supplementary Figure 4). Our results show that the fold parameter does not change the performance significantly. Unless specified, all results reported in the paper were based on 5-fold cross validation.

To determine if the classification performance is sensitive to the choice of classifiers, we used the Random Forest classifier as a comparison. The result demonstrates that the wSM feature set outperforms the SM feature set in terms of both accuracy and AUC (Supplementary Figure 5).

Performance testing using additional external breast cancer datasets (related to Supplemental Figure 6)

To further validate the performance of M-module-based features, we used two external datasets (Supplementary Table 1) that include breast cancer samples of all four stages. We used the same SVM classifiers trained on the TCGA dataset and tested the classification accuracy and AUC of ROC curves on the external datasets.

Construction of meta-network of 4-modules (related to Figure 5A)

Given a module $C_1$with $n_1 = |C_1|$ genes, the module expression profile across k samples is denoted by M with dimension of $(n_1 \times k)$. To summarize the information across genes a module, we obtain the first principle component of M using single value decomposition of M. Briefly, M is decomposed to $M = UDV^T$, where columns of the orthonormal matrices U, V are the left- and right-sigular vectors, respectively. The largest right-singular vector is used as the principle component that represents module C_1. Its dimension is $1 \times k$. Given two 4-modules, we then computed the Pearson correlation between the first principle components of their gene expression profiles. We used the absolute value of the Pearson correlation as the edge weight between the two 4-modules in the meta-network in Figure 5B.

Analysis of feature importance to the classification of specific cancer stages (related to Figure 5B)

Feature weight of a classifier quantifies how discriminative a feature is in a classification task. There are a number of methods to determine the weight of a feature in a classifier. Here, we use the method of sensitivity analysis. During training, the linear SVM minimizes the cost function $J = \|\boldsymbol{w}\|^2/2$. The weight of feature *i* is proportional to the change in the cost function $\Delta J(i)$ that is caused by removing feature *i*, which equals to the absolute value of the weight of feature *i* in the trained classifier, i.e. $|w_i|$ (Boser and Langley, 1992; Cristianini and Shawe-Taylor, 1999; Vapnik, 1998).

Because we train a multi-class SVM classifier using the one-against-one strategy (Hsu and Lin, 2002), there are a total of six feature weights for each module, each of which is obtained from one of the six pairwise classifiers. To determine the importance of a module *i* to the classification of a given cancer stage *j*, we first conduct a min-max normalization of all feature weights in a given pairwise classifier. We call this normalized feature weight w'. We then compute the importance score of feature *i* to stage *j*, RI_{ij}, as following:

$$RI_{ij} = {\sum w'^{j*}_i}\Big/{\sum w'^{**}_i}$$

where w'^{j*}_i is the sum of normalized weights for feature *i* in classifying stage *j* versus the other stages and w'^{**}_i is the sum of normalized weights for feature *i* in all six pairwise classifications.

Comparison of M-modules and 1-modules (related to Supplemental Figure 8 and Discussion)

Compared to the 1-modules, M-modules make use of the information from multiple networks. We hypothesized that M-modules achieve better performance than single-network-based 1-module. To test this hypothesis, we first identified 1-modules in each

network by using the Affinity Propagation (AP) algorithm (Frey and Dueck, 2007) since it has been shown to outperform many other clustering algorithms.

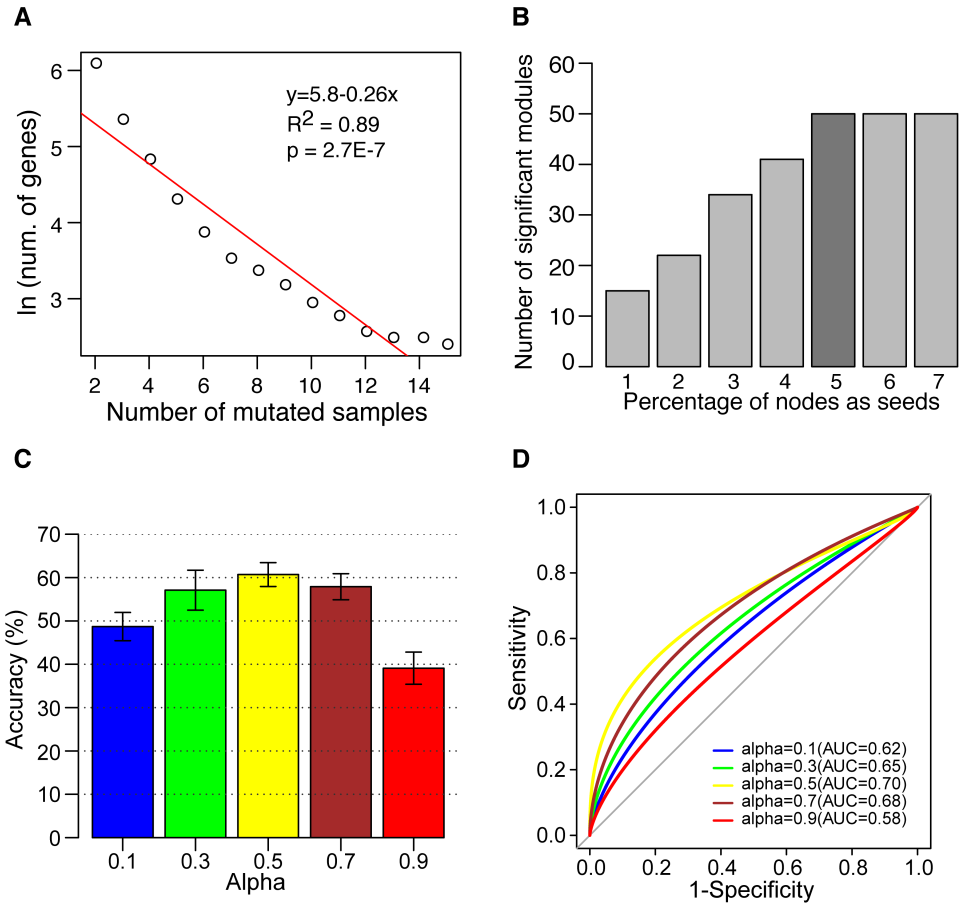
We first compared the two sets of modules in terms of their sensitivity and specificity using the same set of gold-standard pathway annotations used in Figure 2. Second, we used the modules as features for predicting breast cancer stages (Supplementary Figure 2). Finally, because M-module uses graph entropy as its objective function, it is not strictly dependent on graph density and can thus identify sparse modules. Therefore, we compared graph density of the two types of modules.

Functional category analysis of identified gene modules (related to Figures 2 and 5)

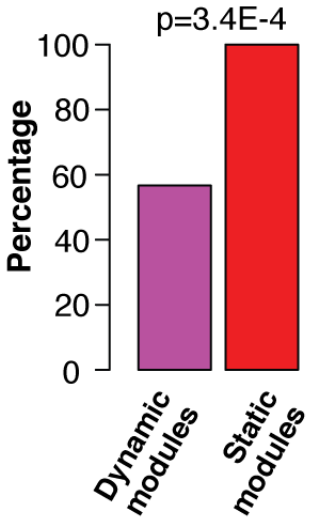
Five sets of reference pathway annotations were used: Gene Ontology(GO) biological process, KEGG pathways, Biocarta pathways, Canonical pathways, and functional gene interactions. functional interactions. Pathway overlap P-values were computed using hypergeometric distribution.

Specificity is defined as the fraction of predicted gene modules that significantly overlaps with reference pathways. Sensitivity is defined as the fraction of reference pathways that significantly overlaps with predicted gene modules. P-values for the difference in specificity and sensitivity were computed using Fisher's exact test. All p-values were corrected for multiple testing using the method of Benjamin-Hochberg.

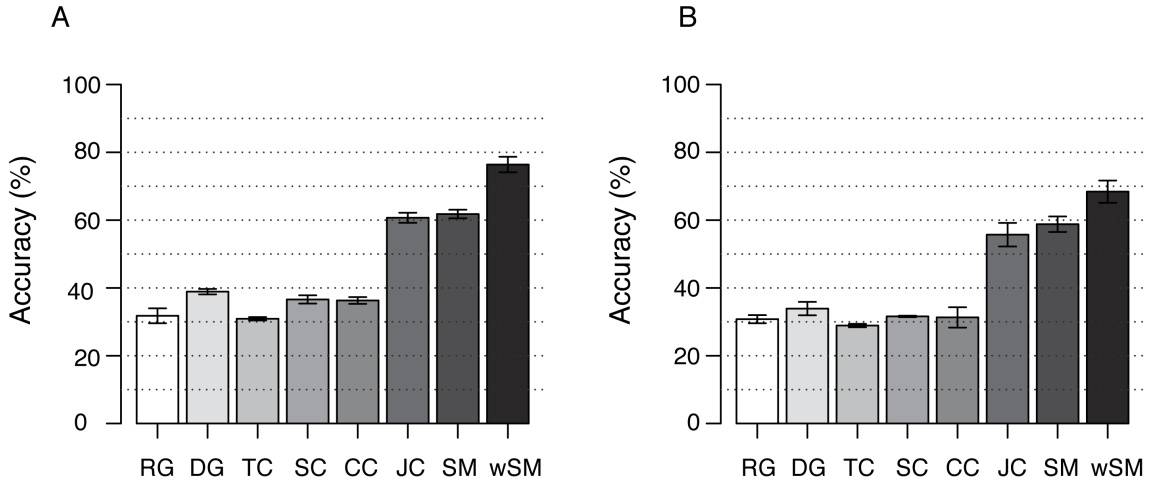
Supplemental Figure 1. *M-module* algorithm parameter tuning. (**A**) Gene mutation frequency distribution. Mutation information is from the COSMIC database. The fitted function is used to compute the prior probability of mutation for a given gene based on the number of observed mutations of the gene in the COSMIC database. (**B**) The number of significant M-modules detected as a function of the number of seeds. **(C-D)** M-module performance as a function of *alpha* value, which determines the relative contributions by network topology and prior probability. **(C)**, Accuracy. **(D)** Receiver operating characteristic curve. We ran *M-module* using different values of alpha (0.1, 0.3, 0.5, 0.7, 0.9). We then used the sets of discovered modules as features to test the performance of breast-cancer-stage prediction.



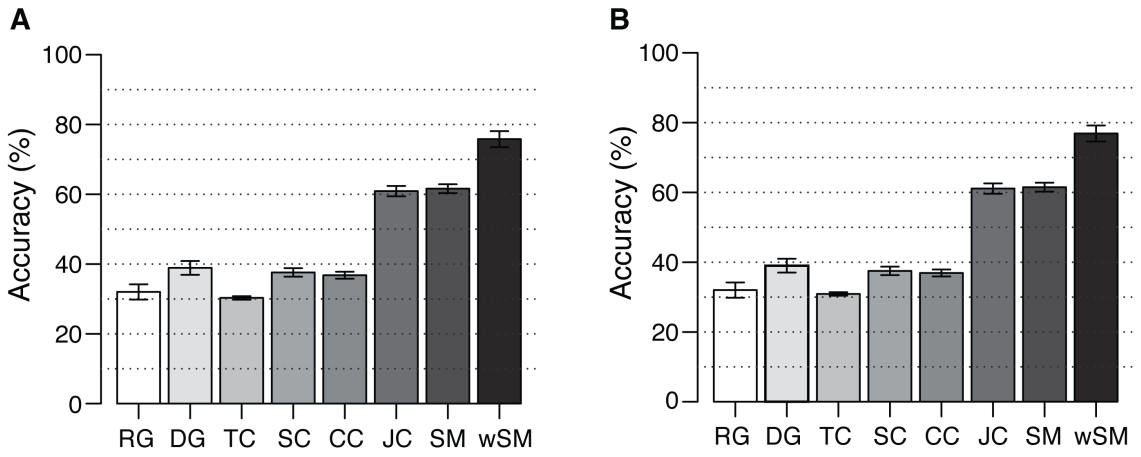
Supplemental Figure 2. Overlap between 4-modules and human protein complexes curated in the CORUM database. Y-axis, percentage of 4-modules that significantly overlaps with manually curated human protein complexes in the CORUM database. Significance of overlap was computed using hypergeometric distribution. P-value for difference in overlap percentage was based on one-sided Fisher's exact test.



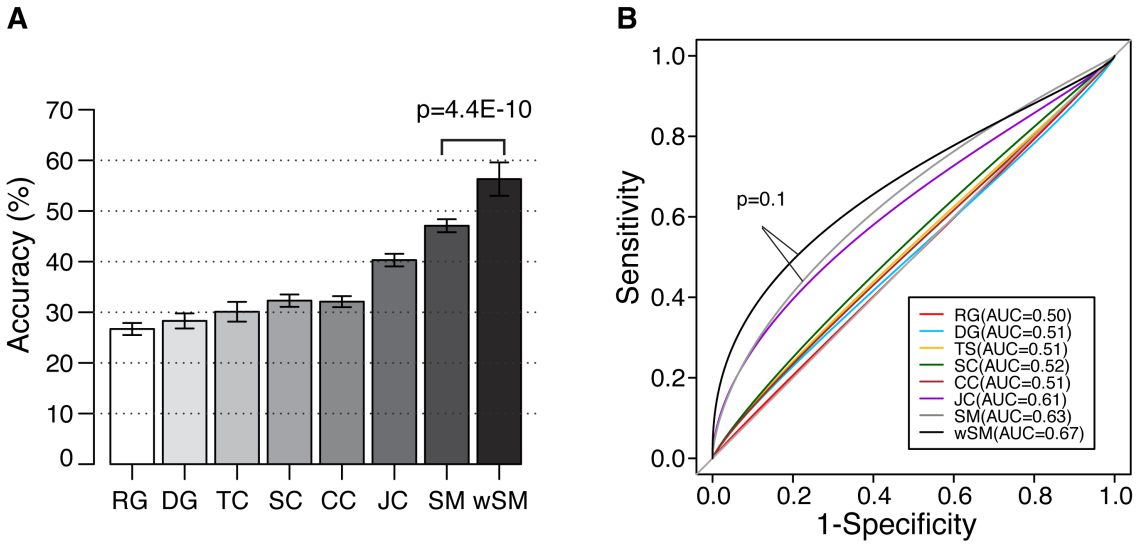
Supplemental Figure 3. Effect of handling of unbalanced data on the classification accuracy of the SVM classifier. TCGA breast cancer data is used. Feature sets are: randomly selected genes (RG, 50 features, 50 genes), differentially expressed genes (DG, 50 features, 50 genes), Tensor Clustering modules (TC, 1573 features, 1601 genes), Spectral Clustering (SC, 91 features, 7737 genes), Consensus Clustering (SC, 100 features, 7737 genes), Jointclustering (JC, 110 features, 7690 genes), significant 4-modules (SM, 50 features, 635 genes), and weighted 4-modules (wSM, 50 features, 635 genes). Y-axis, mean accuracy of 50 independent 5-fold cross validations. Error bar, standard deviation. **A)** Random oversampling strategy for transforming unbalanced to balanced data. **B**) Synthetic sampling strategy for transforming unbalanced to balanced data.



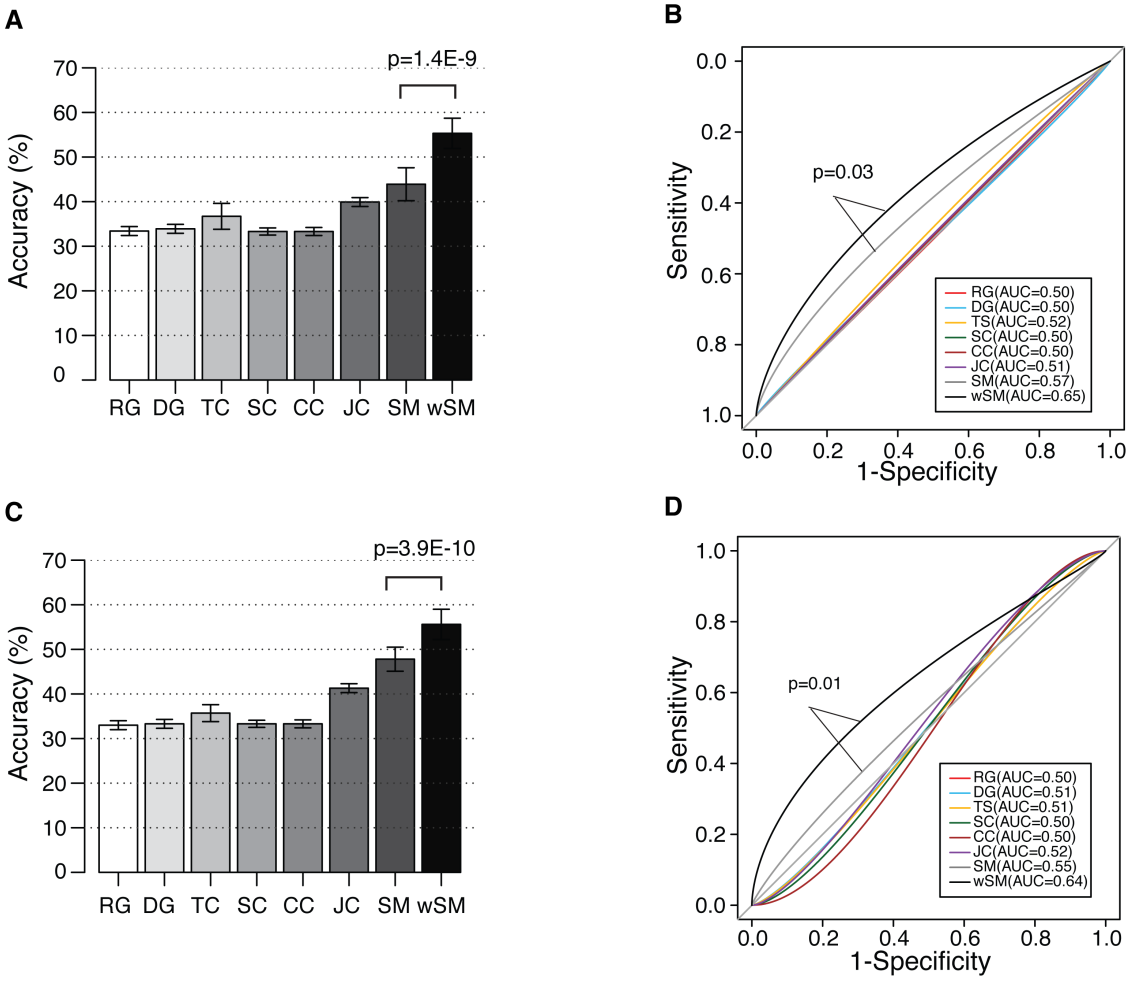
Supplemental Figure 4. Effect of cross validation scheme on the classification accuracy of the SVM classifier. TCGA breast cancer data is used. Feature sets are: randomly selected genes (RG, 50 features, 50 genes), differentially expressed genes (DG, 50 features, 50 genes), Tensor Clustering modules (TC, 1573 features, 1601 genes), Spectral Clustering (SC, 91 features, 7737 genes), Consensus Clustering (SC, 100 features, 7737 genes), Jointclustering (JC, 110 features, 7690 genes), significant 4-modules (SM, 50 features, 635 genes), and weighted 4-modules (wSM, 50 features, 635 genes). Y-axis, mean accuracy of 50 independent 5-fold cross validations. Error bar, standard deviation. **A)** 4-fold cross validation. **B)** 8-fold cross validation.



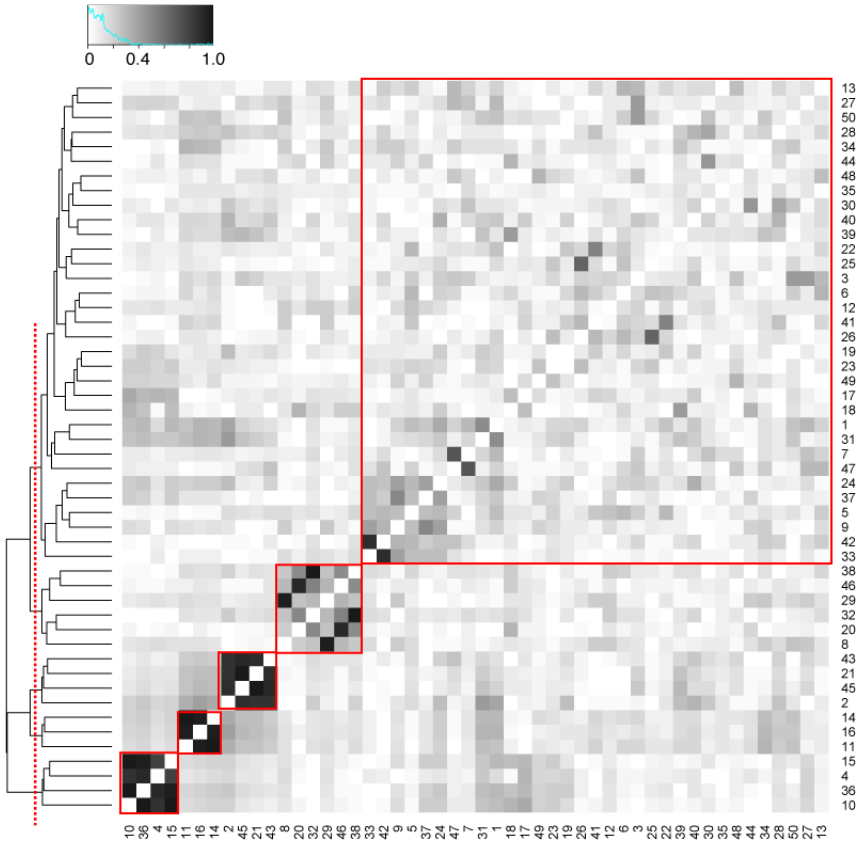
Supplemental Figure 5. Performance of different feature sets using a random forest classifier. Feature sets are: randomly selected genes (RG, 50 features, 50 genes), differentially expressed genes (DG, 50 features, 50 genes), Tensor Clustering modules (TC, 1573 features, 1601 genes), Spectral Clustering (SC, 91 features, 7737 genes), Consensus Clustering (SC, 100 features, 7737 genes), Jointclustering (JC, 110 features, 7690 genes), significant 4-modules (SM, 50 features, 635 genes), and weighted 4-modules (wSM, 50 features, 635 genes). (**A)** Classification accuracy of breast cancer stages. Y-axis, mean accuracy. Error bar, standard deviation. **B)** Receiver operating characteristic curves for random forest classifiers trained with different feature sets. Values in parenthesis are Area Under the Curve. Results in both A) and B) are based on 50 independent 5-fold cross validations.



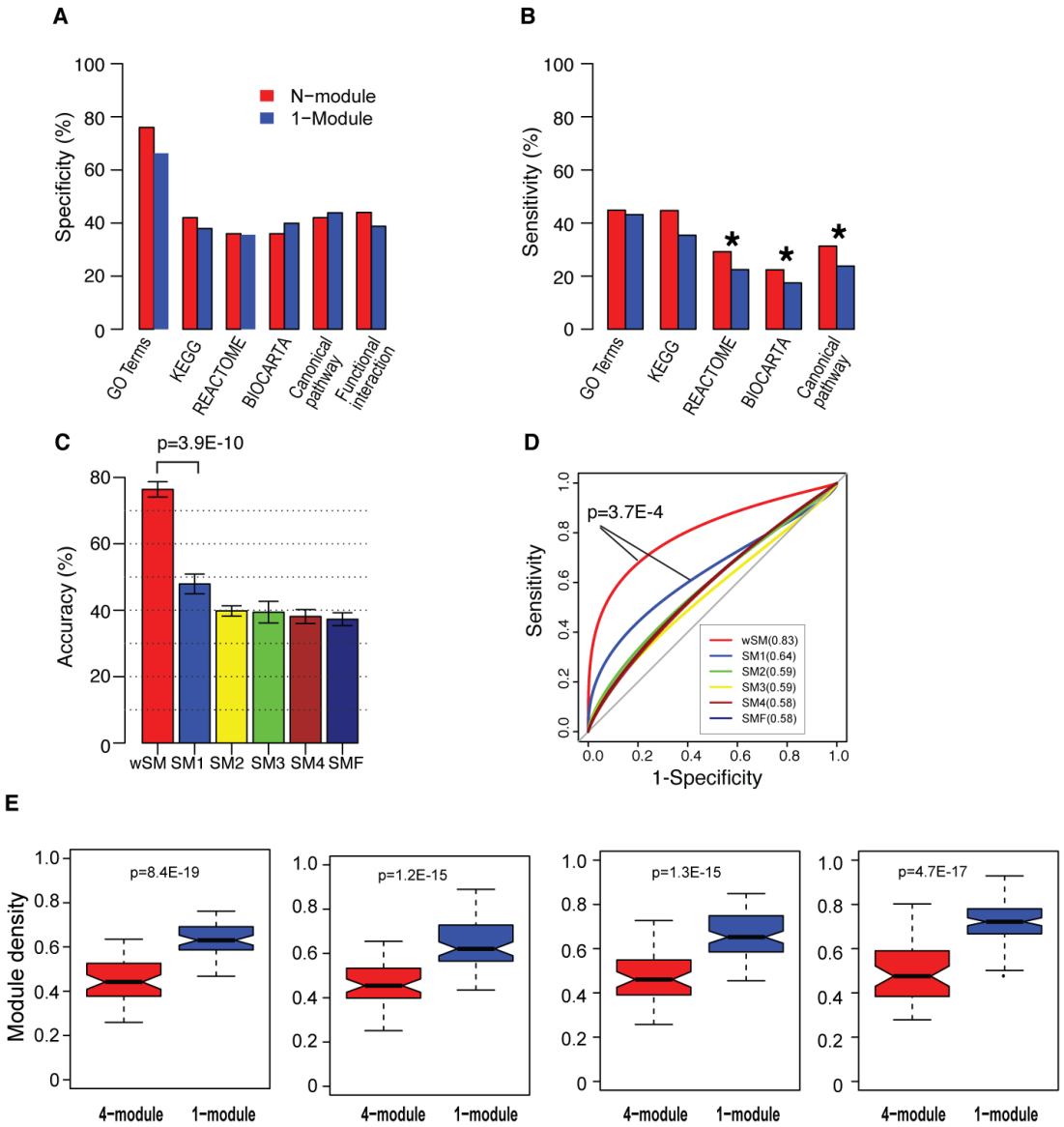
Supplemental Figure 6. Performance of the SVM classifier trained on TCGA dataset and tested on external microarray datasets. Feature sets are: randomly selected genes (RG, 50 features, 50 genes), differentially expressed genes (DG, 50 features, 50 genes), Tensor Clustering modules (TC, 1573 features, 1601 genes), Spectral Clustering (SC, 91 features, 7737 genes), Consensus Clustering (SC, 100 features, 7737 genes), Jointclustering (JC, 110 features, 7690 genes), significant 4-modules (SM, 50 features, 635 genes), and weighted 4-modules (wSM, 50 features, 635 genes). (**A-B**) Accuracy (A) and Receiver Operating Characteristic Curve (B) on the Esserman *et al.* dataset.; (**C-D**) Accuracy (C) and Receiver Operating Characteristic Curve (D) on the Boersma *et al.* dataset.



Supplementary Figure 7. Hierarchical clustering of the 4-modules of the breast cancer data.



Supplementary Figure 8. Comparisons between 4-modules and 1-modules. (A) Specificity of 4-modules and 1-modules. Gene modules are evaluated by a set of gold-standard pathway annotation. Specificity is defined as the fraction of gene modules significantly enriched for genes of some reference sets. **(B)** Sensitivity of the methods. Sensitivity is defined as the fraction of reference sets significantly enriched for genes of some modules found by a method. *, p <0.01. P-values were based on Fisher's exact test. **(C)** Classification accuracy of breast cancer stages using Support Vector Machine (SVM) classifier and 4-module and 1-module feature sets. Y-axis, mean accuracy. Error bar, standard deviation. wSM, connectivity-dynamic-score-weighted 4-modules; SM1-4, 1-modules derived from individual stage 1-4 data. SMF, union of all 1-modules. **(D)** Receiver operating characteristic curves for SVM classifiers trained with different feature sets. Results in both C) and D) are based on 50 independent 5-fold cross validations. **(E)** Module density for 4-modules and 1-modules. P-values were based on one-sided t-test.



Supplemental Tables

Supplemental Table 1. Summary of the breast cancer datasets used in this study.

	Database Source	Accession ID	# Total Samples	# Stage-specific samples
TCGA	TCGA data portal	NA	531	I:92;II:297;III:112;IV:30
Boersma et al.	GEO	GSE5847	47	I:4; II:30; III:12; IV:1
Esserman et al.	GEO	GSE22226	150	I:4; II:63; III:76; IV:3

References

Boersma, B.J.*, et al.* (2008) A stromal gene signature associated with inflammatory breast cancer, *International journal of cancer. Journal international du cancer*, **122**, 1324-1332.

Boser, A. and Langley, P. (1992) An training algorithm for optimal margin classifiers. *Fifth Annual Workshop on Computational Learning Theory*. ACM, Pittsburgh, pp. 144-152.

Boser, B.E., Guyon, I. and Vapnik, V. (1992) A training algorithm for optimal margin classifiers. *In Proceeding of the Fifth Annual Workshop on Computational Learning Theory*. ACM Press, pp. 144-152.

Cristianini, N. and Shawe-Taylor, J. (1999) *An introduction to support vector machines*. Cambridge University Press, MA.

Edge, S.B., DR; Compton, CC; Fritz, AG; Greene, FL; Trotti, A. (2010) *AJCC Cancer Staging Manual*. Springer.

Esserman, L.J.*, et al.* (2012) Chemotherapy response and recurrence-free survival in neoadjuvant breast cancer depends on biomarker profiles: results from the I-SPY 1 TRIAL (CALGB 150007/150012; ACRIN 6657), *Breast cancer research and treatment*, **132**, 1049-1062.

Fawcett, T. (2006) An introduction to ROC analysis, *Pattern Recognition Letters*, **27**, 861-874.

Frey, B.J. and Dueck, D. (2007) Clustering by passing messages between data points, *Science*, **315**, 972-976.

He, H.B. and Garcia, E.A. (2009) Learning from Imbalanced Data, *Ieee T Knowl Data En*, **21**, 1263-1284.

Hsu, C.W. and Lin, C.J. (2002) A comparison of methods for multiclass support vector machines, *Ieee T Neural Networ*, **13**, 415-425.

Lancichinetti, A. and Fortunato, S. (2012) Consensus clustering in complex networks, *Scientific reports*, **2**, 336.

Letunic, I., Doerks, T. and Bork, P. (2012) SMART 7: recent updates to the protein domain annotation resource, *Nucleic acids research*, **40**, D302-305.

Li, W.*, et al.* (2011) Integrative analysis of many weighted co-expression networks using tensor computation, *PLoS computational biology*, **7**, e1001106.

Narayanan, M.*, et al.* (2010) Simultaneous clustering of multiple gene expression and physical interaction datasets, *PLoS computational biology*, **6**, e1000742.

Newman, M.E.J. (2006) Finding community structure in networks using the eigenvectors of matrices, *Phys Rev E*, **74**.

Pleasance, E.D.*, et al.* (2010) A comprehensive catalogue of somatic mutations from a human cancer genome, *Nature*, **463**, 191-196.

Ruepp, A.*, et al.* (2010) CORUM: the comprehensive resource of mammalian protein complexes--2009,

Nucleic acids research, **38**, D497-501.
Tusher, V.G., Tibshirani, R. and Chu, G. (2001) Significance analysis of microarrays applied to the ionizing radiation response, *Proceedings of the National Academy of Sciences of the United States of America*, **98**, 5116-5121.
Vanunu, O*., et al.* (2010) Associating Genes and Protein Complexes with Disease via Network Propagation, *PLoS computational biology*, **6**.
Vapnik, V. (1998) Statistical learning theory. Wiley Interscience.
Watson-Haigh, N.S., Kadarmideen, H.N. and Reverter, A. (2010) PCIT: an R package for weighted co-expression networks based on partial correlation and information theory approach, *Bioinformatics*, **26**, 411-413.