Supplementary Methods

Strains and growth conditions

Microarray expression data for salt stress, oxidative stress and heat shock for all species other than *S. japonicus* and *S. pombe* were collected as described by (Wapinski *et al*. 2010) and are available at GSE38478. The strains, growth conditions and heat shock experiments for *S. japonicus* and *S. pombe* are described below, followed by the microarray hybridization and pre-processing for all species and experiments.

The strains used in the study are described in (Wapinski et al. 2010) with the addition of *S. pombe* 972 h+ and *S. japonicus* IFO1609. Briefly, cultures were grown in the following rich medium (termed BMW): yeast extract (1.5%), peptone (1%), dextrose (2%), SC amino acid mix (Sunrise Science) 2 g/L, adenine 100 mg/L, trptophan 100 mg/L, uracil 100 mg/L. For each strain, cells were plated onto BMW plates from frozen glycerol stocks. After 2 days, cells were taken from plates and re-suspended into liquid BMW, and counted using a Cellometer Auto M10. A 3 mL BMW culture inoculated at $1x10^6$ cells/ml and placed in a New Brunswick Scientific Edison model TC-7 roller drum on the highest speed until saturated (1-2 days). The saturated cultures were then used to inoculate batch cultures in 2 liter Erlenmeyer flasks for the Heat shock experiments described below. Flasks were transferred to New Brunswick Scientific Edison and water bath model C76 shakers set to 200 rpm.

Expression datasets measuring heat shock response in *Schizosaccharomyces*

Cultures for each species were grown in 650 ml of BMW at 22 °C to between 3×10^7 and 1×10^8 cell/mL (OD_{600} = 1.0 for *S. pombe*, and 1.5 for *S. japonicus)*. The shift to heat-shock temperature was carried out as follows by splitting the overnight culture into two 300-ml cultures and collecting cells via vacuum filtration (Nanopore). The cell-containing filters were resuspended in prewarmed media to either control (22 °C) or heat-shock temperatures (37 °C). Density measurements were taken approximately 1 min after cells were resuspended to ensure that concentrations did not change during the transfer from overnight media. A total of 12 ml of culture was harvested 5, 15, 30, and 60 min after resuspension by quenching them in liquid methanol at –40 °C, which was later removed by centrifugation at –9 °C and stored overnight at –80 °C. Cell density measurements were repeatedly taken every 5–15 min for the first 2 hr after treatment. Harvested cells were later washed in RNase-free water and archived in RNAlater (Ambion) for future preparations. Cells were also harvested from cultures just before treatment for use as controls.

RNA preparation, probe labeling, and microarray hybridization

Analysis was carried out as described previously (Wapinski et al. 2010). Briefly, total RNA was isolated using the RNeasy midi or mini kits (Qiagen) according to the provided instructions for mechanical lysis. Samples were quality controlled with the RNA 6000 Nano ll kit of the Bioanalyzer 2100 (Agilent). Total RNA samples were labeled with either Cy3 or Cy5 using a modification of the protocol developed by Joe DeRisi (University of California at San Francisco) and Rosetta Inpharmatics that can be obtained

at http://www.microarrays.org.

Microarray hybridization and data pre-processing was carried out as described previously (Wapinski et al. 2010). Briefly, for each time point, either two or three biological replicates were hybridized with the Log phase sample as the reference in all cases. We used two-color Agilent 55- or 60-mer oligo-arrays in the 4X44 K or 8X15 K format for the *S. cerevisiae* strain (commercial array; four to five probes per target gene) or the custom 8X15 K format for all other species (two probes per target gene). After hybridization and washing per the manufacturer's instructions, arrays were scanned using an Agilent scanner and analyzed with Agilent's Feature Extraction software (release 10.5.1.) The median relative intensities across probes were used to estimate the expression values for each gene per replicate, and these median values across replicates were used to estimate the overall expression response per gene per time point.

Arboretum algorithm details

Arboretum is a model-based clustering approach that uses a probabilistic generative model to analyze multiple expression datasets, one for each species. Each dataset resides at a leaf node (extant species) of a species tree describing the phylogenetic relationships between species. The generative model generates values for two types of random variables: (a) hidden variables representing the module assignments in both ancestral and extant species, and (b) observed variables encoding expression for each gene in a species. The cluster membership is modeled by conditional distributions for every branch of the

species tree, describing the probability of a gene belonging to a cluster in a species given the cluster membership in its immediate ancestor. The expression data at each leaf node is modeled by a Gaussian mixture model. (Since modeling expression at the ancestral nodes requires the inference of additional hidden variables, we restricted ourselves to inferring only module memberships at ancestral nodes). An integral part of Arboretum is that it naturally handles one-to-many mappings of genes over any number of species. This is done by incorporating the gene tree directly inside Arboretum's cluster inference. In the following sections we describe the different parts of the model in detail, inference of cluster assignments and parameter estimation.

Modeling module assignments and their evolution

We assume that every gene in an extant species evolves its module assignment from a single ancestral version, which is present at the LCA (root of the species tree) and let K denote the maximum number of modules that can exist in a species. The LCA has a prior probability distribution, a multinomial, $p(k)$, $1 \leq k \leq K$, which specifies an initial assignment to a module. Every other species t has a module transition matrix, P_t, which relates the modules in species t to modules in t's immediate ancestor. Every element in the transition matrix $P_t(i,j)$ is the conditional probability of a gene to be in module i in species t given that its ancestral gene was in module j in t's immediate ancestor.

The module evolution process generates the module assignment of all genes in an orthogroup, one at a time, using the structure (but not branch lengths) of the gene tree associated with the orthogroup. The structure of the gene tree for an orthogroup with no

duplications or losses ('uniform orthogroup') is the same as the species tree (Wapinski et al. 2007a). To generate the module assignments for an orthogroup, we sample a module assignment from the prior distribution at the LCA, propagating the assignment down the tree via the transition matrices along the branches of the species tree. For example, if a transition matrix has a high value on the diagonal, the gene is more likely to maintain its module assignment at that branch. At the leaf nodes, we generate the expression of a gene from the Gaussian indicated by the propagated module assignment. For a non-uniform orthogroup with a duplication event, we proceed down the tree as in the uniform orthogroup case, until we reach the point of duplication. At the duplication node, we draw two samples from the transition matrix, each of which evolves down the rest of the tree independently following the same procedure as before. Thus the evolution process takes into account the phylogenetic relationships across the species and between orthologs and paralogs. We use this tree structure to devise a tractable module inference procedure.

Modeling observed expression data

The expression data at an extant species t is modeled by a mixture of K Gaussians (Hastie et al. 2003):

$$x_{ti} \sim \sum_{k=1}^{K} \alpha_k \mathrm{N}(\mu_{tk}, \Sigma_{tk})$$

where the k^{th} mixture component describes the expression profile of the k^{th} module, $1 \leq k \leq K$, $\boldsymbol{x_{ti}}$ denotes the expression profiles of the genes, μ_{tk} is a d_t-dimensional mean vector, Σ_{tk} is the diagonal covariance matrix for the k^{th} mixture component, and d_t is the number of measurements for each gene in species t. Note, d_t may be different for different t,

enabling us to handle cases with different number of measurements per species.

Expectation Maximization (EM) framework for model learning

The EM framework for model learning has two steps: *expectation step*, in which hidden variables are inferred from the current model parameters, and *maximization step*, in which parameters are estimated from the expected values of the hidden variables.

Expectation: Inference of module assignments. Let z_i denote the set of hidden variables denoting the module assignments for all genes in the i^{th} orthogroup, G_i. These hidden variables are related via G_i's gene tree such that a gene's module membership in a non-root species t depends upon the gene's module membership in t's immediate ancestor$_i$ Accordingly, $\boldsymbol{z}_i$ is composed of z_{ri}, denoting a gene's module assignment at the LCA, r, and $z_{ti}^{k|k'}$ denoting the conditional membership for all other species t in module k given that its immediate ancestral version is in module k'. Our inference problem is to infer the posterior probability distribution of these hidden variables given the data, $P(\boldsymbol{z}_i|\boldsymbol{x}_i)$, where $\boldsymbol{x}_i$ denotes the measured expression profiles of the genes associated with G_i. Let $\gamma_{ti}^{k|k'}$ specify the posterior probability of t's gene to be in module k, given that the immediate ancestral version of this gene is in module k'. To infer this posterior probability, we make a crucial independence assumption needed to perform tractable inference: the module assignment of a gene at species t depends only upon the subset of the expression data that comes from the subtree below. This allows us to compute the posterior probability at each internal node using computation from its child nodes. Our inference procedure is thus recursive, where the computation we perform at a non-root node, t, to estimate the

posterior probability at that node is used for estimating the normalization constant of t's parent. We begin at the leaf nodes, to estimate the γ's. The product of γ's at two sibling leaf nodes would then give the normalization constant for their immediate common ancestor node. If a node represents a duplication event, because we assume that after duplication the two duplicates evolve independently, the contribution from the sub-tree below the duplication is also a product. Subsequently, we would obtain the normalization constants of an intermediate node by taking the product of its subtrees. When we reach the LCA, the product of the subtrees give the full posterior distribution of the joint of module assignments given the expression data pertinent to the orthogroup.

Maximization: estimation of parameters

The parameters in our model are: (a) module transition probabilities, (b) Gaussian mixture model parameters. These parameters can be estimated in closed form by deriving the expected likelihood with respect to the parameters . The maximum likelihood mean estimate for the j^{th} module of the t^{th} species, μ_{jt}, is very similar to the standard Gaussian mixture model case, except the hidden variables, and takes k^2 rather than k, because of the conditional dependence on the immediate ancestral module:

$$\mu_{tj} = \frac{\sum_i \sum_{l=1}^{K} \gamma_t^{j,l} x_{ti}}{\sum_i \sum_{l=1}^{K} \gamma_t^{j,l}}$$

Here $\gamma_t^{j,l}$ represents the expected value of the joint assignment in module j in child species t, and module l in t's ancestor. Similarly for the variance estimate, we need an additional sum to account for the fact that the module assignment in an extant species is

dependent upon its parents. We assume that the co-variance matrix is diagonal. The transition probabilities for each species is estimated from the expected value of the joint assignment of a child and parent module assignment pair, $P(z^{ti} = k, z^{ui} = k'|\mathbf{x}_i)$, which is $P(z^{ti} = k|z^{ui} = k', \mathbf{x}_i)P(z^{ui} = k'|\mathbf{x}_i)$. Note $P(z^{ti} = k|z^{ui} = k',\mathbf{x}_i) = \gamma^{ti}_{klk}$, which we already have from the expectation step. The marginals $P(z^{ti} = k|\boldsymbol{x}_i)$ are estimated recursively by using the marginal at an ancestor to estimate the joint at a child and then the marginal. We begin at the LCA, where we already have the marginal, descending one level to first estimate the joint and then the marginals, until we reach the leaf nodes.

Learning algorithm

Our learning algorithm begins with an initial clustering assignment obtained from partitioning all orthogroups into *k* partitions. This partitioning can be obtained by randomly splitting the data, or by a clustering algorithm that merges all the species data together into a single vector and clusters these concatenated data into modules. We found the second option to have better results in practice (SR, DAT and AR, unpublished data). The clustering is not expected to be good because orthologous genes may not cluster together across species. The algorithm uses these initial module partitions to seed the parameters values for the Gaussian mixtures. We then repeat the expectation and maximization steps until convergence.

During the first round of EM learning module indices may get permuted, in the sense that the assignment of a gene to a module in a leaf node would not be consistent with its assignment in the ancestor (the phenomenon is unique to the leaves). We take two

measures to avoid this. First, the transition matrices are initialized to have heavy diagonals such that a species has a higher prior probability to conserve a gene's module assignment from its immediate ancestor. Second, we have two rounds of EM. After the first EM training, we check for each gene cases where the gene's module assignment is conserved in all intermediate nodes from a leaf to the root, except at the leaf node. If such a case arises, we swap the probabilities of a gene belonging to the module at the leaf and the rest of the path to the root, and perform another training phase of the EM. This step minimizes 'index flipping' at the leaves, and ensures that all modules of the same index across extant and ancestral species are derived from a single ancestral module. Following this step, modules with the same ID have the highest gene content overlap, as expected.

Determining the number of clusters

We selected the number of modules using a combination of penalized log-likelihood of data per species and manual inspection. First, based on penalized log likelihood the maximum number of modules for any species was k=11 (**Supplementary Fig. 15a, Supplementary Methods**). We use Minimum Description Length (MDL) to define the penalized likelihood: $L\text{-}n_{params}/2log(n_{ogs})$ where n_{params} is the number of free parameters, n_{ogs} is the number of orthogroups and L is the data likelihood. For clustering a dataset per species, we learn a standard Gaussian mixture model with $n_{params} = 2kT$, for the k means and variances for all T time points. For Arboretum the number of free parameters for k modules in an extant species is $2kT+k(k-1)$, the first term corresponding to the Gaussian mixture for the T time points, and the second term to the kXk transition matrix. For an ancestral node other than the root we have $k(k-1)$ parameters. At the root node we have k-

1 parameters for the initial module prior distribution. Thus combining over all species we have $n_{params}=k(2T+k-1)s_e+ (k-1)(s_ak + 1)$, s_e is the number of extant species and s_a is the number of ancestral species other than the root. We next ran Arboretum on the entire 8 species dataset with k=5, 7, 9, 11, 13, and 15 modules and found k=11 to be optimal as well. However, upon manual inspection of the k=11 case, we observed that higher values of k did not produce significantly different expression modules, and were prone to seemingly arbitrary re-assignment of module genes between species, given the very similar expression patterns in 'adjacent' modules. We therefore picked k=5 based on manual inspection of the means of the modules inferred by Arboretum (**Supplementary Fig. 15b**), choosing a number where different modules had clearly distinguishable expression patterns (k=5 for heat stress and k=7 for pan-stress below). Although we computed the penalized log likelihood for the different Arboretum runs, we found that that this was not as informative of the different patterns (**Supplementary Fig. 15c**).

For the *Candida* species, the response to heat shock was measured at both 37°C and 42°C (Wapinski et al. 2010). Modules for *C. albicans*'s under both conditions were similar; we focused on data with 42°C, since this is a stronger and more robust response (Wapinski et al. 2010), as *C. albicans* may be adapted to 37°C because of its role as a commensal human pathogen. For *C. glabrata*, the transition matrix was much more diffused and the modules were much less conserved at 42°C than 37°C, and we picked the latter dataset as a conservative choice. Expression patterns of *S. cerevisiae* ESR induced genes are comparable in 42°C and 37°C for *C. glabrata (***Supplementary Fig. 16**).

Algorithms used to compare against Arboretum

We compared Arboretum to two algorithms, Ortho-seeded species-specific clustering (Waltman et al. 2010) and soft *k*-means clustering (Kuo et al. 2010).

The ortho-seeded species-specific clustering is the most straightforward way of clustering multi-species data, and has been previously used in a bi-clustering context In this approach, we concatenate individual species-specific expression data to generate a new matrix with as many columns as the total number of microarray experiments across all species, and as many rows as there are genes in *S. cerevisiae* and at least one other species, filling in columns due to gene losses using the mean from the other measurements. This concatenated matrix is clustered using a standard Gaussian mixture model followed by another round of clustering on individual species-specific data starting with the modules from the first round of clustering on the merged data.

The soft *k*-means algorithm clusters expression data across multiple species such that the measurement points across the different species are all aligned. (Thus, unlike Aboretum and ortho-seeded clustering it requires matching experiments across species.) The algorithm clusters a concatenated matrix of as many columns as there as experiments in any one species, and as many rows as the sum of the rows in species-specific data matrices. The algorithm uses a soft heuristic, which favors orthologous genes to be in the same expression module. This is done by extending the standard *k*-means objective with a user-defined parameter, $0 \leq \rho \leq 1$, which controls the trade-off between optimizing the traditional *k*-means algorithm and favoring orthologous genes to co-cluster. $\rho=0$ yields the canonical *k*-means algorithm. This approach requires 'matching' experiments across

species.

Measures for comparing Arboretum and other algorithms for module inference

We used four measures to compare the performance of Arboretum to other algorithms.

Module stability. We measured stability of modules by estimating the proportion of gene pairs that co-clustered under different random initializations. We used $r = 20$ different random initializations for each algorithm. Because both Arboretum and ortho-seeded clustering are initialized on modules learned from the merged datasets, this initial clustering could have enabled Arboretum and ortho-seeded clustering to infer more stable modules. However, we found that irrespective of whether Arboretum (and ortho-seeded) clustering was initialized on modules from a merged dataset or not, both approaches identified more stable modules, with Arboretum outperforming the ortho-seeded clustering. We computed these stability measures for different species subsets and observed a similar stability performance.

Expression coherence. We measured expression coherence in each module as the average proportion of genes whose expression profiles had a high (>0.8) correlation with the module's mean. We computed this metric for different random initializations of each algorithm and obtained a mean and standard deviation of the module coherence.

Conservation of gene content across species. We estimate the extent of gene content conservation between modules from different species by considering modules in one

species, *s*, at a time and comparing with all the other species. To measure conservation of modules for a pair of species, *s* and *t*, we first pair one module from *s* to a module of *t* using maximal overlap of orthologs based on the Hyper-geometric *p*-value. Conservation of gene content for modules from *s* and *t* is defined as the average of the maximal overlap scores between *s* and *t*. Conservation of gene content for a species s is defined as:

$C_s = \sum_{1 \le t \le S, t \ne s}^{S} \frac{C_{st}}{S-1}$ where C_{st} is the conservation score between species *s* and *t*'s modules, and *S* is the total number of species.

We measure orthology overlap for the i^{th} module from species *s* with n_i^s genes and the j^{th} module with n_j^t genes from species *t*, with n_{ij}^{st} genes in common as the average of the negative logarithm of two *p*-values; one considering the total number of genes in *s* as the background, and one considering the total number of genes in *t* as the background.

Performance on (simulated) ground truth. We used the same simulated data used to study Arboretum parameters above to assess how well other algorithms infer modules in extant species. For soft *k*-means clustering we considered different values of the parameter (ρ) that controls the extent of supporting orthologous genes to cluster in the same module, estimated module match with each of these settings and used the module match that was the highest. Similarly, for Arboretum we considered different values of the parameter initializing the transition matrix and used the highest module match.

Accuracy and sensitivity analysis of initial parameter settings of Arboretum

We examined the ability of Arboretum to reconstruct modules and parameters as a

function of different initial parameter settings using simulated data for which we already knew what the true modules were ('ground truth'). To generate the ground truth modules and parameters, we learned module parameters on the heat shock data of eight species, followed by sampling data from the model using the learned parameters. We used this sampled data as input to Arboretum and inferred modules. Then we compared these inferred modules from Arboretum to the ground truth modules (**Supplementary Fig. 3**), and the inferred transition matrices to the ground truth transition matrices (**Supplementary Fig. 4**).

In particular, we examined the performance of Arboretum by these measures at different values of the 'self transition probability' $p \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$, the user-defined parameter controlling the initialization of the transition matrix. For each value of p, we performed 20 runs each with a different random initialization to also examine how Arboretum's performance depends on the initial means and variance parameters. We defined a module similarity score for each species based on an *F*-score overlap (described below in GO process conservation section). This is obtained by matching each module in the inferred set to a module in the ground truth set based on the maximal match of gene content (as defined by *F*-score), followed by taking an average across the modules.

We found that the accuracy of module assignment was not sensitive to the initial value of p (**Supplementary Fig 3**), was highly accurate at the leaf nodes (80-95%) and more modestly accurate at ancestors (40-65%), with decreasing accuracy the more ancient the

ancestor. Most assignment errors were due to re-assignments to a 'neighboring' module (*e.g.*, a gene assigned to module 4 in the ground truth, is assigned to module 5 in the inferred module, **Supplementary Fig. 5**). Furthermore, errors in the lower nodes would contribute to errors in the higher (more ancestral) nodes, resulting in decreasing accuracy with increasing distance from the leaf nodes. The estimation of transition matrices was highly accurate (mean squared error between the true and inferred transition matrices close to zero for most cases), and did not depend on the initial value of *p* (**Supplementary Fig. 4**).

Enrichment analysis of Gene Ontology (GO) processes and cis-regulatory elements

We use the FDR corrected hyper-geometric p-value to assess enrichment of GO processes in a given gene set. We use the GO terms for *S. cerevisiae* downloaded from the Saccharomyces Genome Database (SGD) Release version 1.1556. For all other species, we use orthology to transfer the Gene Ontology annotations, as previously described (Wapinski et al. 2007b).

For cis-regulatory elements we use a database of species-specific motifs to search for *cis*-regulatory elements in 600 bp upstream of the start codon (Habib et al. 2012). Enrichment is assessed based on the *p*-value from the Hyper-geometric distribution. The species-specific motif library is created by starting from known position weight matrices in *S. cerevisiae* and refined using an expectation maximization framework on individual species sequences.

Comparison to the *S. cerevisiae* Environmental Stress Response (ESR)

We traced the evolutionary history of the ESR induced and repressed genes (Gasch et al. 2000), using Arboretum module assignments inferred under heat shock. Because identifying orthologs in gene families with many duplication events is less reliable (Wapinski et al. 2007), we first analyzed a smaller set of the original induced and repressed genes in *S. cerevisiae*'s ESR program (**Fig. 4a**) that belonged to gene families with at most one duplication event.

To infer the ancestral ESR we used the combined expression data from three stresses and five species and included orthogroups with any number of gene duplications, only requiring that the orthogroup have a gene member in *S. cerevisiae* and at least one other species. We used the LCA modules 1 and 2 to define the repressed Ancestral genes, and the LCA modules 6 and 7 to define the induced Ancestral ESR genes and tested for overlap of these modules with the induced and repressed ESR genes in *S. cerevisiae*, assessing significance using the Hyper-geometric test.

Details of Module Contraction and Expansion Index

A module, *m,* could change in gene content at a phylogenetic point, *s*, in two ways: (**a**) module contraction: genes that are in module *m* in *s*'s ancestor switch to a different module in *s*, and (**b**) module expansion: new genes that were not in *m* join *m* at *s*. To assess module contraction and expansion at each phylogenetic point we estimate a Module Contraction Index and a Module Expansion Index at each species with an ancestor. At each of these phylogenetic points, denoted by *s*, we estimate three counts for

each module *m*: (1) c*onserved pairs*, the number of cases where the module assignment of a gene is *m* in both *s* and its ancestor, (2) *expansions:* the number of cases where module assignment in *s* is *m* but not in ancestor, (3) *contractions*: the number of cases where module assignment in *s*'s ancestor is *m* but not in *s*.

We define the Module Contraction Index (MCI) for module *m* at a phylogenetic point *s*, as the ratio of the number of contractions divided by the number of genes in module *m* in *s*'s ancestor *t*. We define the Module Expansion Index (MEI) at *s* for module *m* as the number of expansions divided by total number of genes in module *m* in *s*. Thus contractions are defined with respect module size in the ancestral species, and expansions are defined with respect to the module size in the child species.

We also define a global MCI of a module *m* as the sum of contractions for that module across all species with a parent (that is except the LCA) divided by a normalization term, Z_m^c, defined as follows: $\sum_{s,t \in S, s \neq t} N_{st}^m$, where *S* is the set of all species other than the LCA, *t* is *s*'s immediate parent, and N_{st}^m is the number of genes for which we have a module assignment in both *s* and *t* and the module assignment of the gene is *m* in the ancestor *t*.

Similarly, we define MEI as the sum of all expansions divided by Z_m^e defined as $\sum_{s,t \in S, s \neq t} M_{st}^m$, where M_{st}^m is the number of genes for which we have module assignments in both *s* and *t*, but the module assignment of the gene is *m* in the child *s*.

Comparing the re-assignment tendency of genes under different responses

We use the inferred ancestral module assignments to estimate the number of times a gene is reassigned at any phylogenetic point starting from the LCA to any of the leaf nodes.

We handle orthogroups with and without duplications separately. For orthogroups without duplication events, the re-assignment fraction is simply the number of reassignments for the gene divided by the number of phylogenetic points at which the gene is present (not lost). For orthogroups with duplications, we compute the re-assignment fraction pre- and post-duplication separately. The pre-duplication re-assignment fraction is the same as in the orthogroups without duplications. Post-duplication, we average the two reassignment fractions from the two copies of the gene. Finally, the reassignment fraction of the entire orthogroup is an average of the pre and post-duplication re-assignment values. We classify a gene to be "high mobility" if it has a re-assignment score of ≥0.5 or more re-assignments, and "low mobility" or "stationary" if it has a re-assignment score of <0.05 or less. We chose these cut offs based on the shape of the cumulative distribution of the number of re-assignments.

Assessing GO process conservation and divergence

To assess the extent of orthologous gene content conservation for processes enriched in modules of the same IDs across species, we use an *F*-score based overlap for the modules for a pair of species, considering only those processes enriched in extant species. F-score similarity for a pair of gene sets G_1 and G_2, with the set $\frac{G_{12}}{G_2} G_{12}$ of genes in common is defined as $F = \frac{2 * P * R}{P + R}$, where P is defined as precision, $\frac{|G_{12}|}{|G_1|}$ and R is defined recall, . F-score is a number between 0 (no overlap) and 1 (complete overlap). Let a process p be enriched in a set of extant species S_i in module i and in set S_j in module j. To compute the conservation of gene content for same module IDs, we take an average of F-scores

first over each set S_i and S_j, and then over modules i and j. To compute the gene content conservation for different module IDs, i.e., between modules i and j, we take the average F-score between all pairs of species s and t, where $s \in S_i$ and $t \in S_j$. To identify representative examples of processes that are conserved in gene content we used a cut off of F-score >0.8 for processes associated with modules with same IDs, and >0.7 for processes associated with modules of different IDs. To identify examples of processes that are not conserved in gene content, we used a cut off of F-score <0.4 for processes associated with modules of the same IDs, and F-score <0.3 for processes associated with modules of different IDs. These thresholds were selected to capture processes in the top and bottom 10% of the cumulative distributions.

REFERENCES

Habib N, Wapinski I, Margalit H, Regev A, Friedman N. 2012. A functional selection model explains evolutionary robustness despite plasticity in regulatory networks. *Mol Syst Biol* **8**: –.

Hastie T, Tibshirani R, Friedman JH. 2003. *The Elements of Statistical Learning*. Corrected. Springer.

Kuo D, Licon K, Bandyopadhyay S, Chuang R, Luo C, Catalana J, Ravasi T, Tan K, Ideker T. 2010. Coevolution within a transcriptional network by compensatory trans and cis mutations. *Genome Res* **20**: 1672–1678.

Waltman P, Kacmarczyk T, Bate A, Kearns D, Reiss D, Eichenberger P, Bonneau R. 2010. Multi-species integrative biclustering. *Genome Biol* **11**: R96.

Wapinski I, Pfeffer A, Friedman N, Regev A. 2007a. Automatic genome-wide reconstruction of phylogenetic gene trees. *Bioinformatics* **23**: i549–58.

Wapinski I, Pfeffer A, Friedman N, Regev A. 2007b. Natural history and evolutionary principles of gene duplication in fungi. *Nature* **449**: 54–61.

Wapinski I, Pfiffner J, French C, Socha A, Thompson DA, Regev A. 2010. Gene duplication and the evolution of ribosomal protein gene regulation in yeast.

Proceedings of the National Academy of Sciences **107**: 5505–5510.