Data preparation

Data sources for the different species is shown in Appendix I [*include a 'head' of the 3 files*]. In the GO-terms file, the associations were labelled as "valid" if, for at least on of the association available in data, they correspond to Experimental evidence scores (precisely: 'EXP', 'IDA', 'IEP', 'IMP', 'IPI', 'IGI') and to the category of Biological process (BP), and as "non-valid" otherwise. In order to maintain this distinction throughout the analysis between "valid" and "non-valid" associations these two groups of associations were up-propagated independently and then the files were combined keeping the label. The R function "uppropagate.r" was used for up-propagating.

Domain and GO-terms files were pruned to exclude genes that are not available in the network file, as a requirement for the BMRF code. Then, the GO-size filter was applied to exclude the GO-terms that were too general or whose number of known associated genes was excessively low for the BMRF computations (Appendix I). The GO-size filter is based solely on the "valid" associations for two reasons: (1) the non-valid associations are not used in the validation and (2) the GO-size filter allows to make sure that there are enough number of genes in the validation. Analogous to the GO-size filter, BMRF uses a DF-size filter to exclude from the analysis the domains whose number of genes is below a certain threshold.

Part 1 – Evaluation of the BMRF method

BMRF [1] was used to do PFP and learn about the impact of different method and network parameters on the prediction performance. In this framework, predictions are made individually for each GO-term that passes the GO-size filter (see Appendix I - Concepts). The predictions, however are not entirely independent for each GO-term in the dataset because the genes that are not associated with any of the GO-terms are treated differently. These genes are coded as "unkowns", and the number of unkowns genes depends on the number of GO-terms in the database (which can be regulated through the GO-size filter). Each gene in the network file will enter the BMRF code with one of three possible labels: '1', '0' or '-1', and it will be predicted as '1' or '0', where '1' stands for positive, '0' stands for unlabeled (it is not known whether the gene has the function) and '-1' stands for unknown (it is not known whether the gene has the function and its label in the training set will be '0' or '1' based on Gibb-sampling taking into account the label of its neighbors). Thus, although the genes enter the BMRF code with three labels, the training set consists only of two labels.

The advantage of treating the unknown genes (genes known to be associated with zero GO-terms) differently is that genes that have never been predicted as positives are less likely to be non-associated for a given function than those genes that have been found as positives for some function but not for the function of interest. This has to do with the fact that some function are more difficult to predict than others. Thus, if based on data, a gene has never been identified as positive for any function, it is more fair to assume that the function is particularly difficult to predict using experimental approaches, that assuming that a very low number of genes have the function. In other words, a large portion of unlabeled genes ("0") implies that the function is rare, (almost never observed), whereas a large portion of -1s implies that the function is difficult to predict. This distinction between unlabeled and unknowns genes, additionally, allows to do predictions in a fairer fashion. By treating the gene of interest as unknown (-1) instead of as unlabeled (0), the prediction of the gene-GO association we are interested in, will be more free from the prediction of the GO-term. If the portion of "-1" however becomes very large with respect to the portion of "0", however, Gibb-sampling will fail in the relabeling, because it will expect that the portion of genes that have the function is very large.

The labelling is as follows:

gene	label	label test	change with fold
Positives labelled as "non-valid"	1*		
unkowns	-1		
Positive in test-set	-1	1	YES
Negative in test-set	-1	0	YES
Positives in train-set	1		YES
Negatives in train-set	0		YES

Table 1: Labelling of genes in BMRF

** Only take value “1” if the parameter Only_EES is set to “False”, otherwise these genes are excluded from the analysis.*

The validation was as follows: k-fold 10, was used and only the associations coded as "valid" entered the test set. Genes classified as unknowns were also excluded form the test set. The k-folds were made independently for the positive set and the unknowns but with a similar value of k (k:10 in part 1 and k:2 in part 2). Because the allocation of genes to folds is a random process, 20 replicates we carried.

Analysis of the impact of the quality of data in PFP

In order to investigate the impact that the quality of the data has on the predictions, a portion of associations were randomly removed. For this, we distinguished between 2 types of associations: association of the GO-term of interest and associations of other GO-terms.

Similarly, the prediction performance was computed for different portions of the edges that were removed from the anlaysis. For this, we distinguishes four types of edges: edges positive -positive, epn, enn, te. Portions subtracted were[...]

Finally, we carried analysis adding random associations between genes and the GO-term of interest. We did not sample genes from the set of genes that are positives for the GO-term but not validated. Portions of noisy associations added were[…]

In order to compute the correlation between different parameters that may be affecting the accuracy of prediction, the following information was extracted:

For each gene-GO association:
-number of edges
-number of edges positives-positives
-number of edges positives-negatives
-number of edges negatives-negatives

For each GO-term:
-depth, using the function 'getAllBPChildren" from the R-packe GO.db
-number of genes
-number of validated labels
-sum of the number of edges, of the genes associated with the GO-term
-sum of the number of edges positives-positives, of the genes associated with the GO-term
-sum of the number of edges positives-negatives, of the genes associated with the GO-term
-sum of the number of edges negatives-negatives, of the genes associated with the GO-term

For each gene:
-number of GO-terms

By looking at the values of correlations, as well as the differences in data and prediction performance between the different species we identified the information feature that more directly affect the prediction performance. Based on this, we chose an approach to improve the BMRF PFP method. Parameters defined to evaluate the BMRF are given in Appendix I- Concepts.

The analysis were carried out for k-fold:10 and only validated associations entered the test-set. Because the assignation of genes to the folds is a random process we carried the analysis with 20 replicates and averaged the results. The standard deviation across folds within the same replicate, as well as across replicates, were computed.

Part 2- Development of PU-BMRF

Similarly to Part 1, in Part 2, the folds for the set of positives were created solely based on the validated associations. However, for simplicity, in part 2 the non-validated positives associations were always excluded from the analysis. This can be done by simply setting the “only_EES” parameter to “True”.

Steps 1 to 6 aim the computation 86 features including 70 non-GO-specific features and 16 GO-specific features These features wre computed for each of the 1,656,000 total gene-associations (138 GO-terms x 12,000 genes). Note that, for chickens, only 138 GO-terms passed the GO-size filter.

- **Step 1 – Similarity Matrix:**

A similarity matrix between the GO-terms is computed using the R package “GOSim”. The use of computing this matrix is two folded: First, it allow to extract a set of unrelated GO-terms, in case we cannot carry the analysis for all the GO-terms that pass the filter (for instance, due to time constrains); and second, this matrix will be used in the computation of the features (step 5).

- **Step 2 – Creating the folds:**

The training and test-sets are created by randomly sampling genes among the positive associations for each GO-term. Then, for each fold, one GO_file is created, in which the associations in the test-set have been exuded. Also, the set of genes that, after “hidding” the test set, are associated with the GO-terms are stored, and also their neighbors. Thus, for each GO-term, two objects are stored: the set of positive and the neighbours of the genes that aree in the positives. Finally, another object is extracted with the neighbors of each gene. These objects are different for each fold and will be used in steps 4 and 5.

- **Step 3 – Network features:**

Transitivity, closeness and betweeness are computed for each gene-GO combination in three different networks:

- A network for all edges that have at least one node in the set of Positives. This is expected to be useful because two types of nodes are included (positive and unlabeled), and genes that are positive (either discovered or to be discovered) are expected to be more interconnected in this network.
- A network or all edges that have at least one node in the set of Positives, and all the edges that have both their nodes in the set of neighbors of positives (step 2). In this network it is also expected that the positive (either discovered or to be discovered) are more interconnected than the genes that are not associated with the GO-term.
- A network of all edges except those that link to genes in the positive set. We expect that in this network, the genes that are positive are less interconnected.

- **Step 4 – non-GO-specific features:**

The following features are computed for each of the 12,000 genes:

- features f1-f4 refer to the number of GO-terms of the gene. It is expected that gene that are associated with a large number of GO-terms are more likely to be associated with a novel GO-term. We expect this probability to be higher for genes that are associated with a large number of specific GO-terms (before up-propagating) because they are more likely to be involved in regulatory functions. Genes that are associated with a large number of GO-terms only after up-propagating, however, may be associated with these simply because they are associated with more general GO-terms.

f1) The number of GO-terms the gene is associated with.
f2) The number of GO-terms the gene is validated-associated with
f3) The number of GO-terms the gene is associated with, in a GO file before up-propagating
f4) The number of GO-terms the gene is associated with, in a GO file before up-propagating

- features f5 and f6 refer to the number of GO-terms of the neighbors of the gene.

f5) The sum of the GO-terms that are associated with the genes that are co-expressed with the target gene.
f6) Thes um of the number of GO-terms that are associated with the genes that are co-expressed with the target gene in a database where only validated GO tersm are considered.

- features f7-f9 refer to the number of neighbor.

f7) The number of genes that are co-expressed with the gene of interest. BMRF accounts for this information, but only when the Pearson Correlation threshold was set to 0.35.
f8) The number of genes that are co-expressed with the gene of interest and are associated with at least 2 GO-terms.
f9) The number of genes that are co-expressed with the gene of interest and are associated with at least 5 GO-terms.

We expect that genes that are co-expressed with genes that have multiple functions are more likely to have multiple functions and therefore are more likely to be associated with a novel GO-terms. The thresholds of 2 GO-terms and 5 GO-terms in f8 and f9 were chosen based on the variability of the feature. We are interested in features that are highly variable across the genes.

- features f10 to f70 are same as f1-f7 but for different features Pearson correlation thresholds. Mainly: 0.1, 0.2, 0.35, 0.5, 0.6, 0.7 and 0.8. Note that as the network database changes, so does the GO-file because BMRF does not allow for any gene that is not in the network. Due to the constrains in the data that come after the different correlation thresholds, it is expected that if a gene is associated with a very large number of GO-terms when the Pearson correlation was high (i.e. 0.6), it must be a gene involved in many different functions, whereas it may be that other genes are associated with more GO-terms when the Pearson correlation is lower, and this should also be considered.

The possibilities of restricting the data-set based on the Pearson correlation cutoff greatly increases the information available. This is because based on different correlation cutoffs we may be be able to observe different patterns in data. Note that this is particularly useful since we are using a condition-independent network where the data of different experiments is combined.

- **Step 5 – GO-specific features:**

For each gene, we computed up to 16 GO-specific features. First, we defined four intersection for each gene and most of the features defined in step 5 will be computed for each of these intersection (for each gene).

Intersections of genes:

(1) Whether the gene of interst is in the set of positives
(2) Genes that are found in the neighbors of the gene of interets and in the set of positives
(3) Genes that are found in the gene of interets and the neighbors of the genes in the set of positives
(4) Genes that are found in the neighbors of the gene of interest and the neighbors of the genes in the set of positives.

F1-f4) Following from step 3, for each gene, we compute the sum of the betweeness, transitivity and closeness of the GO-terms that are in the interactions 1-4.

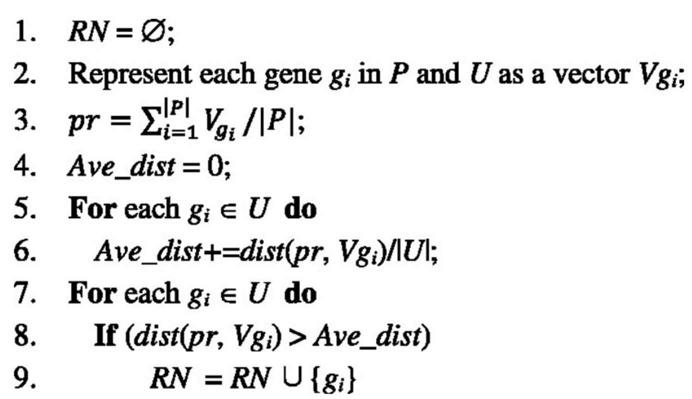
f5-f9) The number of genes in intersections 1-4 and the portions of neighbors of the genes of interest that are in the intersection s1-4.

F10-f13) The number of domains of the genes in intersections 1-4 the number of genes that share domains in intersections 1-4, and the number or unique domains that are sheared between genes in intersections 1-4.

f14-f16) The number of genes in interaction 1-2 (neighbors of the genes in the positive set are not considered here) weighted by the degree of similarity between the GO-terms they have in common and the GO-term we are interested in.

- **Step 6 – extraction of RN:**

The databases with features information obtained in steps 4 and 5 were combined and the values of the features were scaled. Thus, for each GO-term we have a database with 86 features per gene. Checking the label of the genes that are in the training set, we apply the algorithm:



```
1. $RN = \varnothing$;
2. Represent each gene $g_i$ in $P$ and $U$ as a vector $Vg_i$;
3. $pr = \sum_{i=1}^{|P|} V_{g_i} / |P|$;
4. $Ave\_dist = 0$;
5. For each $g_i \in U$ do
6.     $Ave\_dist += dist(pr, Vg_i)/|U|$;
7. For each $g_i \in U$ do
8.     If ($dist(pr, Vg_i) > Ave\_dist$)
9.         $RN = RN \cup \{g_i\}$
```

Illustration 1: Algorithm for extracion of RN. Source: [1]

We check whether any of the RN is in the set of non-validated positive cases and we removed from the analysis those that did so. We tried different thresholds in step 6 of the algorithm in Illustration 1 (default value is 1) and through an iterative process we adjusted the threshold to the highest value that allows to extract a maximum number of RN. We gradually increased the threshold by 0.05 if the criteria was not satisfied (thus, if the number of RN was excessively high for our purpose). We proceeded the analysis for different values of “maximum number of RN”, mainly, for 1000, 2000… and 8000.

In step 6, we also extracted the same amount of RN but through random extraction and stored them separately.
Two approaches were also used to extracted RN. Instead of defining a fixed number of RN, we allowed the threshold to change according to a desired value of AUC in the process of extraction of RN. For instance, we can specify that we want to extract as many RN as possible as long as the AUC is equal to 1 in the process of extraction. The genes that are used to evaluate the performance of extraction are those in the test-set. These genes were excluded from the analysis in step 1 of the PU-BMRF and therefore were not considered to define the threshold in step 6 of the algorithm in illustration 1.

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3467748/