Appendix III – Additional results

Part 1- Choice of the network data and tune the model parameters for prediction performance using BMRF.

- **Section (a) Impact of the co-expression threshold on the prediction performance**

Conventionally a Pearson Correlation of 0.7 is used as a threshold for co-expression analysis. However, in the case of Chickens, using a Pearson correlation of 0.7, leads to an excessively low number of validated associations. Subsequently, only 9 GO term pass the GO-size filter used by BMRF (see appendix I-Concepts). Furthermore, we observed that the number of GO terms that passed the filter did not improve much when the the GO-size filter was adjusted to include in the analysis GO terms with a low number of associated genes (for minGOsize=0.8, 52 GO passed the filters). We therefore investigated whether by lowering the Pearson correlation threshold we obtained better prediction performance. We choose different co-expression threshold values and we computed AUC, standard deviation across replicates for a given GO -term and number of gene sin network and GO term sin analysis.

Illustration 1 shows the prediction performance (AUC) for chickens using different Pearson correlation thresholds. The distribution of the AUC is hyperbolic, showing its maximum at a Pearson correlation of 0.35.

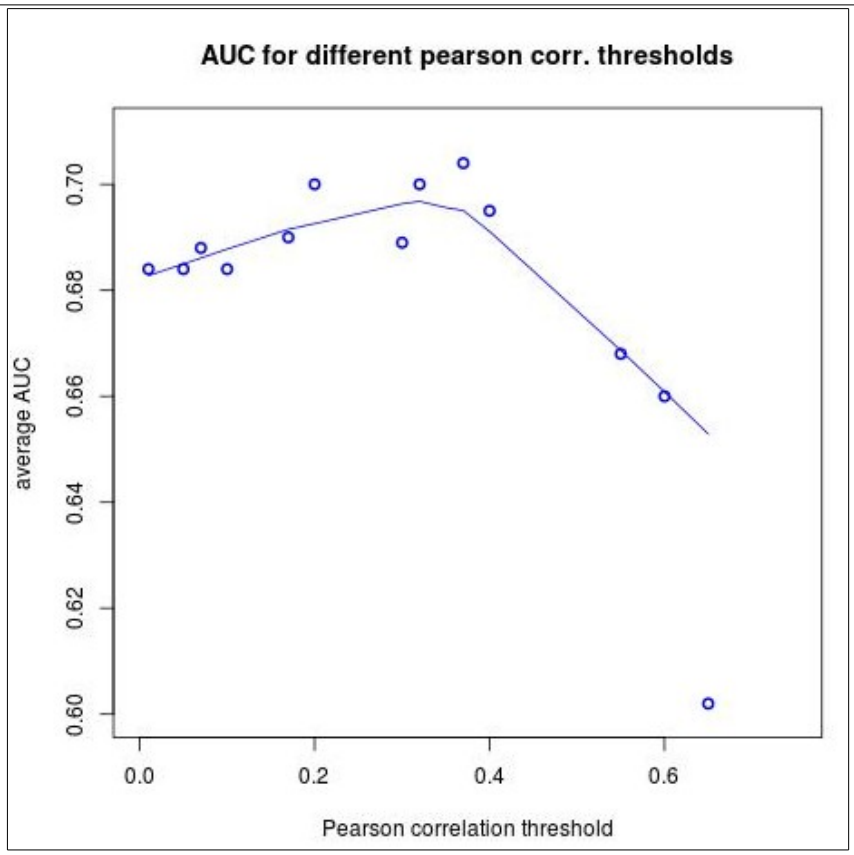


Illustration 1: AUC for different Pearson correlation thresholds for the chickens' co-expression network.

The Y-axis corresponds to the average AUC for all GO-terms considered in the analysis. The line corresponds to the lowness line. Values corresponding to the plot are in table 1.

The network changes considerably base don the co-expresion threshold. Two important parameters that change based on this are the number of genes and the standard deviation across replicates for any given GO-term, as shown in Illustrations 2 and 3, respectivelly.

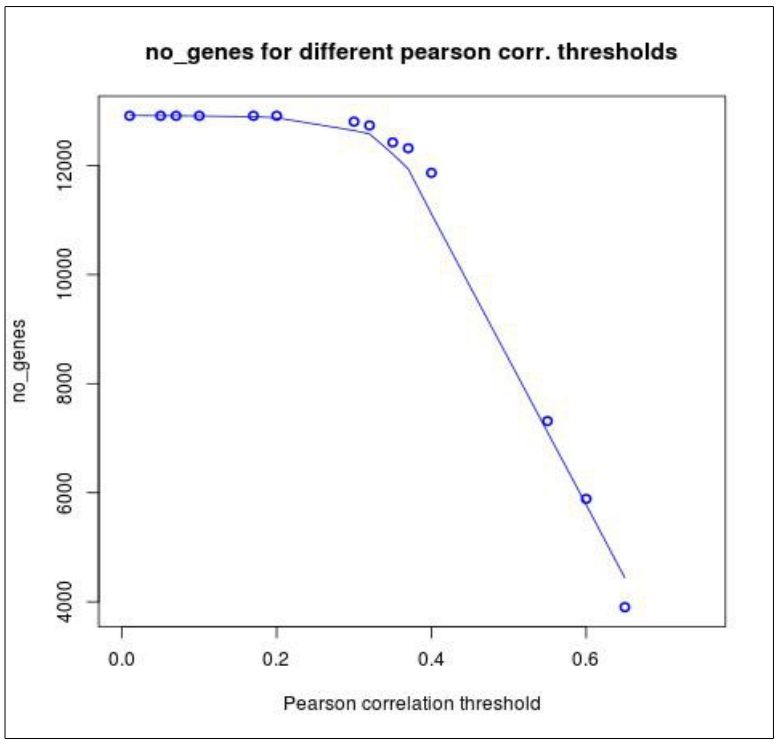


Illustration 2: Number of genes in the network for different co-expression thresholds.

Values corresponding to the plot are in table 1.

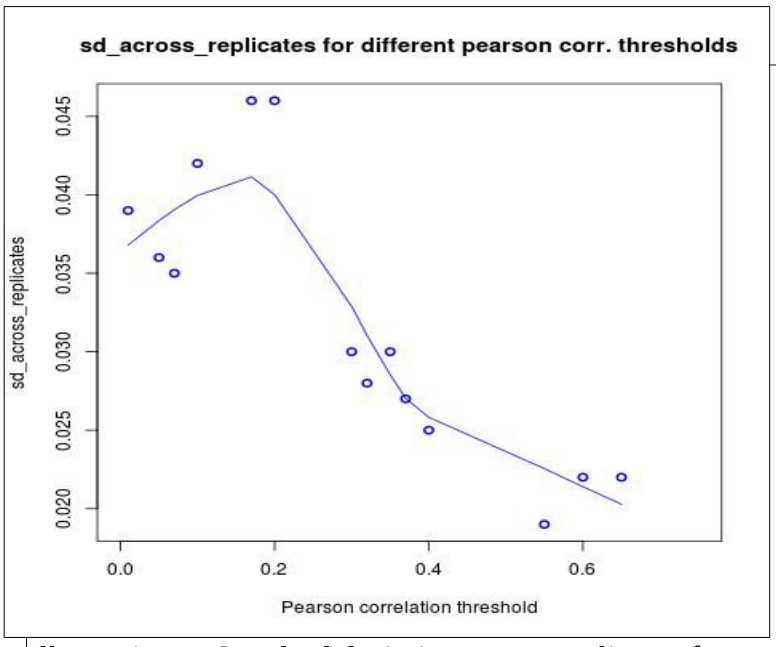


Illustration 3: Standard deviation across replicates for differnet co-expression thresholds.

Values corresponding to the plot are in table 1.

Illustrations 2 show a decrease in the number of genes as the threshold for co-expression becomes more strict. The decrease becomes more sharp when the threshold is above 0.4. However, this may depend on the characteristics of the co-expression analysis.

In illustration 3 we observe that the standard deviation across replicated decreases also as we become more strict in the co-expression threshold. Thus, choosing a high co-expression threshold allows for better reproducibility. On the left hand side of the illustration, we observe that the standard deviation was lower for Pearson correlation thresholds (0.01, 0.02 and 0.015), than for a Person correlation of 0.17 and 0.2. This effect is counter-intuitive, but does not seem to be significant and may be an artifact of the analysis.

Illustrations 1-3 are based on the results in Table 1.

Pearson corr. Threshold	#GO terms pass normal filter	#genes in network	AUC	sd AUC	Mean of SD across replicates
0.7	9	2,784	0.714	0.065	0.022
0.65	21	3,901	0.603	0.064	0.022
0.6	33	5,887	0.660	0.055	0.019
0.55	54	7,314	0.668	0.063	0.022
0.4	134	11,866	0.695	0.056	0.025
0.37	140	12,317	0.704	0.056	0.027
0.35	**138**	**12,424**	**0.726**	**0.080**	**0.030**
0.32	148	12,735	0.700	0.052	0.028
0.3	148	12,805	0.689	0.054	0.030
0.2	150	12,912	0.700	0.068	0.046
0.1	150	12,912	0.684	0.068	0.042
0.07	150	12,912	0.688	0.075	0.035
0.05	150	12,912	0.684	0.074	0.036
0.01	150	12,912	0.684	0.078	0.039

Table 1: Prediction performance choosing different Pearson correlation thresholds, for chicken data. In bold the threshold for which a highest AUC was achieved.

Predictions were highest for a Pearson correlation threshold of 0.35. In this study we used a Conditional independent network to obtain reproducible results. However, It should be considered that this value depends on the characteristics of the co-expression analysis. Furthermore, this value may depend on the species. For instance, in species with a large number of validated data, it is expected that the best possible threshold is higher, given that overall quality of the data is higher; whereas in a species in which the portion of validated data is low (as is the case for chickens), including data of relative quality may be helpful. In other words, the quality of the data may become more important once the criteria for minimum of data required has been satisfied.

- **Section (b). Impact of the different model parameters on the prediction performance**

In this section we investigate the impact in the prediction performance of the number of replicates of analysis, the GO-size filters, the number of k in the k-fold validation and the number of iteration in the Gibbs-sampling. Also we investigated the effect of adding non-validated data and/or domain information.

- Number of replicates

We considered as reproducible, results with less than 0.02 standard deviations in AUC across replicates. Note that the standard deviation that is given in most of the tables, for instance in tables 3 and 4, refer to the standard deviation across GO-terms rather than across replicates of the same GO term.

We investigated how many replicates were required to achieve an average standard deviation of 0.02 across replicates. For this, we carried different runs, of 10 or 20 replicates each and we computed the standard deviation across runs. We observed that the standard deviation across replicates was 0.006 lower when 20 replicates were used instead of 10. Furthermore, we observed that 20 replicates were required to achieve an average stabndrad deviation across replicates below 0.02. We therefore decided to use 20 replicates for each analysis, except for part 3, that we used only 4 replicates due to time constrains.

In another analysis we showed that the standard deviation across 5 runs of 20 replicates was slightly lower for a GO-size filter of (20,0.1) than for (5,0.9): 0.008 vs 0.01, respectively. This makes sense since the standard deviation is slightly larger for those GO terms with fewer genes and more of these GO terms were considered in the analysis when the GO-size filter was (5,0.9) (less strict).

- GO-size filter

Default value for maxGOsize was 0.9, however, for this thesis, we are not interested in predictions for the most general GO terms and we chose value 0.1. A preliminary analysis carried on yeast co-expression data showed that there is not significant increase in the accuracy of prediction when 0.1 was used instead of 0.9. In this analysis, AUC was 0.779 with a minGOsize of 0.1 and 0.775 for 0.9. The standard deviations being 0.08 in both cases. The same analysis showed how the data changes with the GO-size filter (table 2)

scenarios			**data**			
scenario name	**Min GO-size**	**Max GO-size**	**Network size (#conn)**	**#unkown genes***	**#assoc.**	**#GO-terms**
normal	20	0.1	598,174	655	132,249	1,104
default**	20	0.9	598,174	4	264,279	1,187
more GO-terms	10	0.9	598,174	4	273,977	1,738
Only large GO-terms	30	0.07	598,174	688	104,582	832

Table 2: Effect of the GO-size filter on the data

We then investigated the effect of the GO-size filter in the 3 species considered and for yeast ppi.

For this, we carried the analysis in three scenarios with different GO-size filters. We called “normal scenario” to the analysis in which minGOsize was 0.1 and minGOsize:20. We considered that by changing the minGOsize to 9, we are adding to the analysis more specific GO-terms and by removing the filter on maxGOsize (maxGOsize=1) we are allowing for more general GO terms. The lowest value used for minGOsize was 9 because values below this give problems in the computation of the sparse matrices.

	# GO-terms		
scenario	**MinGOsize:9 maxGOsize:0.1**	**MinGOsize:20 maxGOsize=0.1**	**MinGOsize:20 maxGOsize=1**
Description scenario	**Adding more specific GO-terms**	**Normal**	**Adding more general GO-terms**
humans	3328	1982	2069
Chickens0.35	307	138	138
yeast	1772	1104	1187
yeast PPI	1734	1057	1153

Table 3: Impact of GO-size filter in the prediction performance

From table 3, we learn that the number of GO-terms after passing the filter was still low for chickens (307 GO-terms) when minGOsize was set to 9. Due to time constrains we will carry the analysis for the 138 GO terms in chickens when the GO-size filter is (20,0.1). The analysis, however, could be extended to 307 GO terms if the filter was changed to (9,0.1).

	Average AUC (sd)		
scenario	**MinGOsize:9 maxGOsize:0.1**	**MinGOsize:20 maxGOsize=0.1**	**MinGOsize:20 maxGOsize=1**
Description scenario	**Adding more specific GO-terms**	**Normal**	**Adding more general GO-terms**
humans	0.701 (0.017)	0.701 (0.017)	0.703 (0.017)
Chickens0.35	0.726 (0.08)	0.726(0.08)	0.726 (0.08)
yeast	0.757 (0.023)	0.764 (0.016)	0.761 (0.015)
yeast PPI	0.707 (0.028)	0.714 (0.02)	0.712 (0.018)

Table 4: Impact of GO-size filter in the prediction performance

From table 3, we learn that in the case of humans and chickens, the effect of the GO-size filter on the prediction performance was almost null (for the GO-size filters considered). For yeast and yeast PPI, the impact was very small, leading to slightly better performance when the filter was “normal”. Intuitively we would expect a better performance in the scenario “Including more General GO-terms”. However, we did not observe so. This could be regarded as an indication that the relationship between the specificity of the GO-term and the prediction performance is not linear.

This increase in AUC may come because predictions are made for a different subsets of GO terms. For instance, more strict filter leads less GO terms passing the filter. An additional analysis, however, showed that the prediction of individual GO terms are not affected by the GO-size filter. We carried applied BMRF on human data with minGOsize:20, 150 and 400, and we computed AUC only on those GO terms that were predicted with the three filters. The mean AUC (and sd) were:: 0.693 (0.05), 0.693 (0.05) and 0.692(0.05), respectively, indicating that the prediction performance of a GO term is independent on how many GO terms are considered. Table xx in this appendix, however we observe dthat the predictions are, howver, not independt on the number of labelled

genes of other GO terms in the data differnet form th etarget GO terms.

- Number of folds in k-validation

The number of folds in the k-validation is an important model parameter because the association data is highly unbalanced (for each GO term, the number of unlabeled genes is much larger than the number of labeled genes), and therefore low values of k may result in a inadequate use of the train set (using less labeled genes than are actually available), and high values of k may result in an inaccurate prediction performance because AUC may be estimated based on a excessively low number of labeled genes. Table 4 shows the results when we carried analysis using different values of k.

	Average AUC (sd)			
K-fold	**2**	**5**	**10**	**20**
humans	0.667 (0.034)	0.694 (0.021)	0.701 (0.017)	0.705 (0.01)
Chickens0.35	0.7 (0.083)	0.718 (0.082)	0.726(0.08)	0.75 (0.066)

Table 5: Impact of the number of folds in the prediction performance.

Table 4 shows that in human data k:10 is sufficient to achieve the highest possible prediction performance, whereas in chickens, the prediction performance using k:20 instead of k:10 is slightly larger. This not surprising since in chickens the number of labeled genes per GO term is much lower than in humans and a higher value of k is translated in train set with more labeled genes and better prediction performance. In humans, however, the train set seems to have already a large number of positive cases when k is equal to 10. We will choose the value 10 for the model parameter 'k', because larger values imply that the number of positive cases in the training set may not be sufficient (at least for some of the folds), and the estimates of accuracy of prediction become less reproducible.

- Number of iteration in Gibb-sampling

In BMRF, Gibb sampling is used to estimate the label of the unknown genes (a definition of unkown genes is given in Appendix I-concepts. We investigated whether more iterations are required when the number of unknown genes was large (default value of GS is 30 iterations). We carried analysis when the number of unknown genes was above 3000 (this was achieved with a minGOsize of 400), and we estimated AUC when GS was 30 and 500. We observed that the mean AUC and the mean standard deviation across replicates were near identical in both analysis. We therefore concluded that increasing the number of GS iterations is not helpful when the number of unknown genes becomes very large. We therefore chose value for the number of GS iterations 30 in all analysis.

- Non-validated data and domain information

We investigated the effect of adding domain information and non-validated GOterm-gene associations in the analysis. Table 6 shoes the results in three scenarios defined based on the information used.

	Average AUC (sd)		
scenario	**Not including information**	**Including domain information**	**Including both, domain info. and non-valid info. (normal approach)**
humans	0.654 (0.027)	0.701 (0.017)	0.705 (0.017)
Chickens0.35	0.57 (0.068)	0.724 (0.08)	0.726(0.08)
yeast	0.773 (0.093)	0.792 (0.089)	0.764 (0.016)
yeast PPI	0.73 (0.103)	0.747 (0.0992)	0.714 (0.02)

Table 6: Impact of domain info. and non-valid data on the prediction performance.

In table 6, we would intuitively expect an increasing AUC from the left of the table to the right (as more information was included in the analysis). However, in the case of yeast and yeast PPI, we observed the best prediction performance when the domain information was included but not the non-valid info. Furthermore, predictions were better when non of the sources of information was included (left) than when both sources were considered. This is a clear indicator that in the case of yeast, and yeast_ppi, the non-valid information worsens the prediction performance. A possible explanation for this is that in the case of yeast, a very large portion of the gene-function associations are validated and therefore including the non validated information increases the noise without a corresponding increase in the accuracy of prediction.

In the case of chickens and humans, the domain information is more relevant than for yeast and yeast_ppi, accounting for roughly 5% higher AUC. Whereas the non-valid information slightly helps in chickens and humans. An explanation could be that in the absence of enough validated data for humans and chickens, adding non-validated information may add noise but it also improves the resources in the network method.

Part 2- Prediction with BMRF

In this part we get a better insight on the prediction performance for the species considered using BMRF. Illustration4 and 5 show the distribution of AUC for the different GO terms, and the standard deviation, respectively.

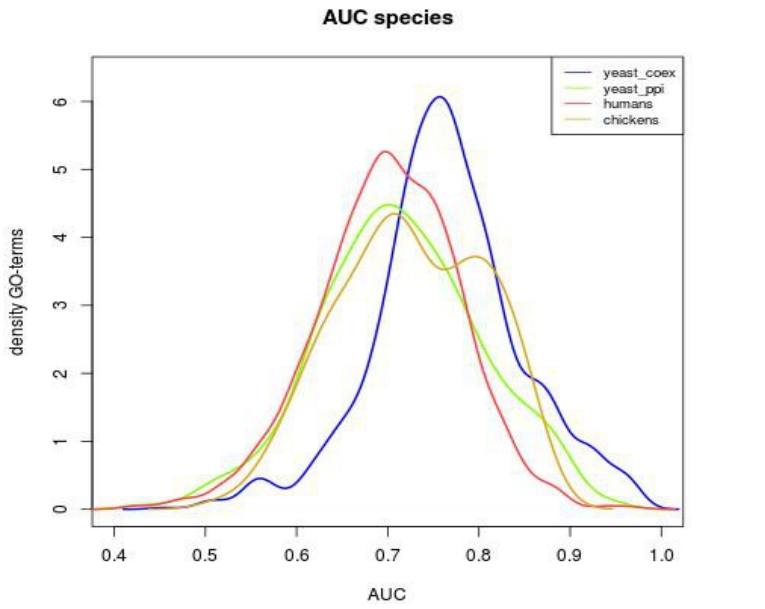


Illustration 4: Distribution of AUC for the different GO-terms in the different species.

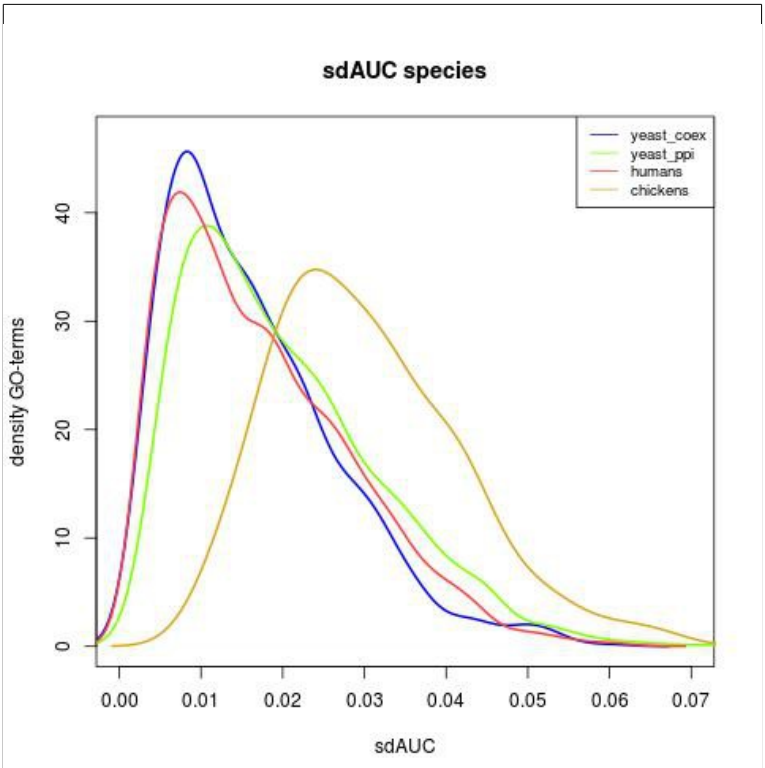


Illustration 5: Distribution of the standard deviation of AUC across replicates for the different GO-terms in the different species.

From illustration 4, we observe that AUC is marginally larger for yeast and that the highest value of AUC is translated in an overall incerase of AUC in the individual GO terms. Prediction perfomance for yeast_ppi, humnas anc hcikens is more similar, and seem sto be larer for chichekns than fo rhumans and for humans than for yeast_ppi. We also observe that the right side of the curce decreases more sharply for choickens than in the other 3 cases.

Illustration 5 shows a similar distribution of standtad deviation across replicates for the different GO terms in humans, yeast amnd yeast_ppi, whereas for chckens the standr deviation is considerably larger. This was expeted since the number of genes that are associated with the GO terms is much lower fo this species (Illustration 2 -Appendix II-Overview of data) and a lower number of positive scases is expected to be associated wit ha higher standrad deviation, because the training set is more unblanced and the predictionwill depend on which of the postive genes enter the trainng roi the test set.

To get an overview of how the AUC of specific GO terms change depending on thespecies considerd, we ploted the AUC of 20 randomly chose GO terms that were predicted in the four cases (illustration 7).

We then got an overbiew of which portion of the GO terms are abve a certain AUC value in the differnet cases

	yeast_ppi	yeast_co	humans	Chicken_07	Chicken_035
number of GOS with AUC>0.6	1013 (92%)	1155 (97.3%)	1819 (92%)	8 (89%)	140 (99.29)
mean depth	6.2	6.0	6.1	2.4	4.0
sd depth	1.5	1.6	1.5	0.5	1.3
>0.7	723 (66%)	1016 (85.6%)	1200 (60.06%)	5 (56%)	113(80.14%)
mean depth	6.5	6.0	6.2	2.4	4.1
sd depth	1.5	1.6	1.6	0.5	1.3
>0.8	281 (25,6%)	428 (36%)	384 (19.8%)	2	44(31.2)
mean depth	7.0	6.5	6.4	2.5	4.3
sd depth	1.4	1.4	1.7	0.7	1.4
>0.9	54 (5%)	83 (7%)	158 (8%)	0	0
mean depth	8.2	7.3	7.4	NaN	NaN
sd depth	1.0	1.4	2.1	NA	NA
>0.95	37 (3.4%)	21 (1.77%)	143 (7.2%)	0	0
mean depth	8.0	8.1	6.3	NaN	NaN
sd depth	0.9	1.7	1.5	NA	NA

Table 7: Portion of GO terms above different AUC thresholds and their depth

In table 7 we learn that although AUC I large for chickens than for other species, predictions do not reach accuracy above 90% for any of the GO terms. This can be regarded as, BMRF is less not completely satisfactory for poorly annotated species, such as chickens and needs some improvement.

Chicken_07	Chicken_035	Chicken_035_filter8
9	142	347
0.728	0.762	0.754
0.062	0.077	0.093
0.718	0.771	0.765

Table 8: Prediction perfomance for chickens in three sceraios depending on co-expresion thershodl and GO-size filter

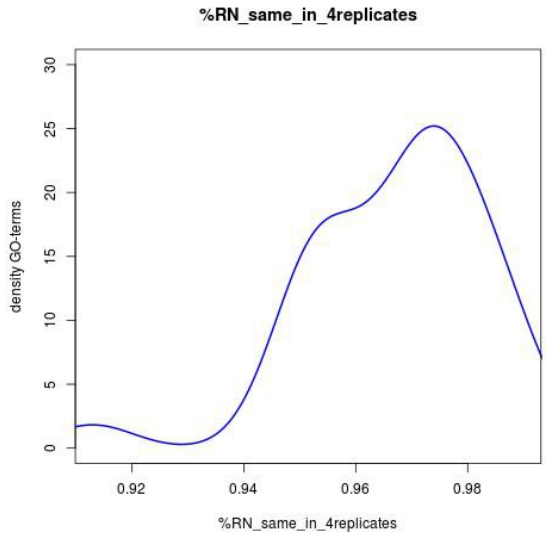


Illustration 6: Reproducibility in the process of extraction of RN when a maximum of 3000 RN were exctracted

Illustration 6 shows a more visual rep[resentation of the prediction performance in the species considered.

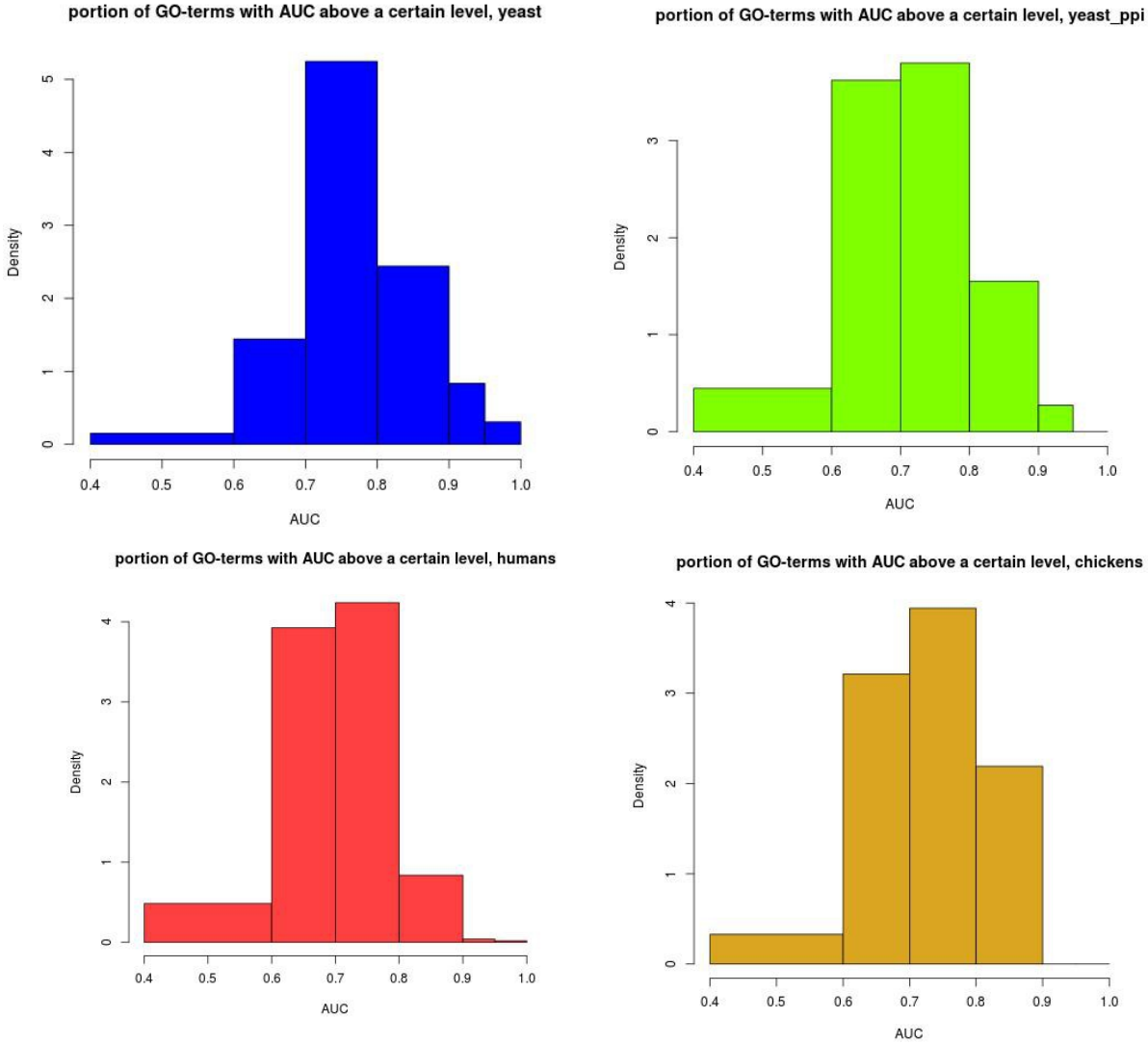


Illustration 6, shows once more that predictions are better for yeast, the nchieksn and yeast_ppi and finally humans. Howeverm, we observe that whereas for yeas_ppi there are a few eGO terms whose accuracy is above 0.9, this is not the case for any of the Go terms predictens with chicken data. It is interesting to investigate how the histogram will change when PU-BMRF is used.

		AUC(sd)	
#RN	**BMRF**	**BMRF random extract**	**BMRF-PU**
1000		0.643 (0.094)	0.723 (0.08)
2000		0.65 (0.09)	0.75 (0.072)
3000		**0.64 (0.1)**	**0.758(0.084)**
4000		0.636 (0.095)	0.751(0.086)
5000		0.652 (0.087)	0.725 (0.094)
6000		0.644 (0.087)	0.728 (0.089)
7000		0.638 (0.096)	0.716 (0.092)
8000		0.636 (0.095)	0.701 (0.095)
all	0.706 (0.0793)		

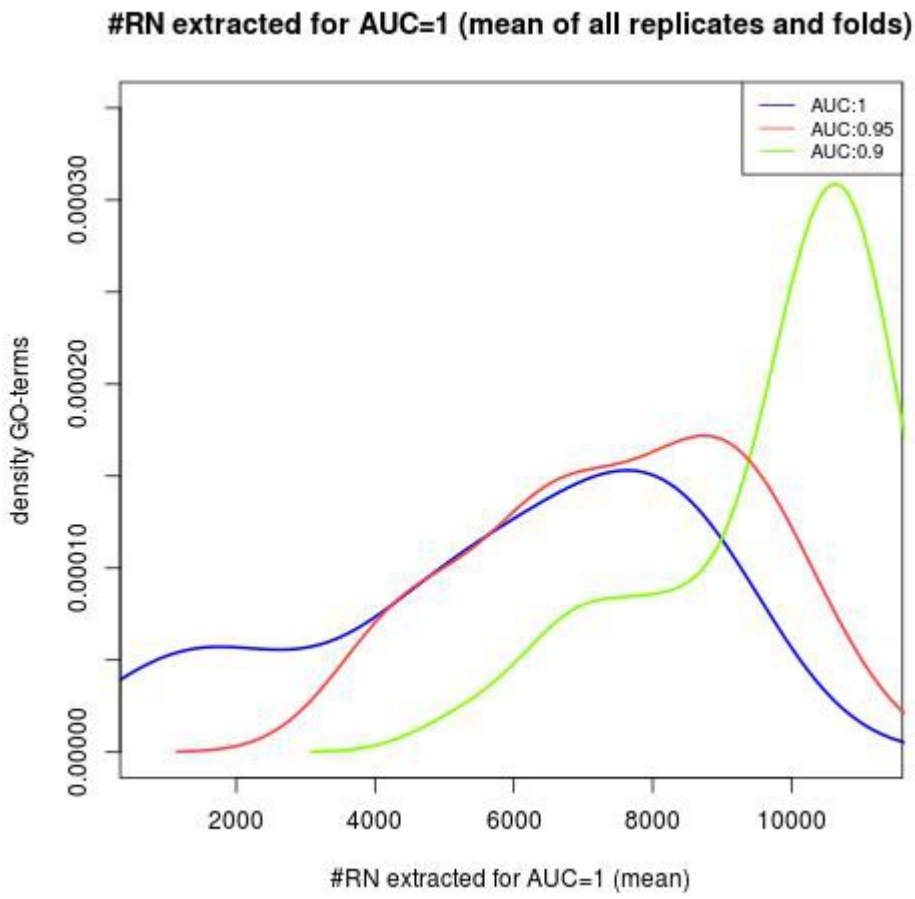
Table 9: Comparison accuracy of prediction BMRF vs PU-BMRF

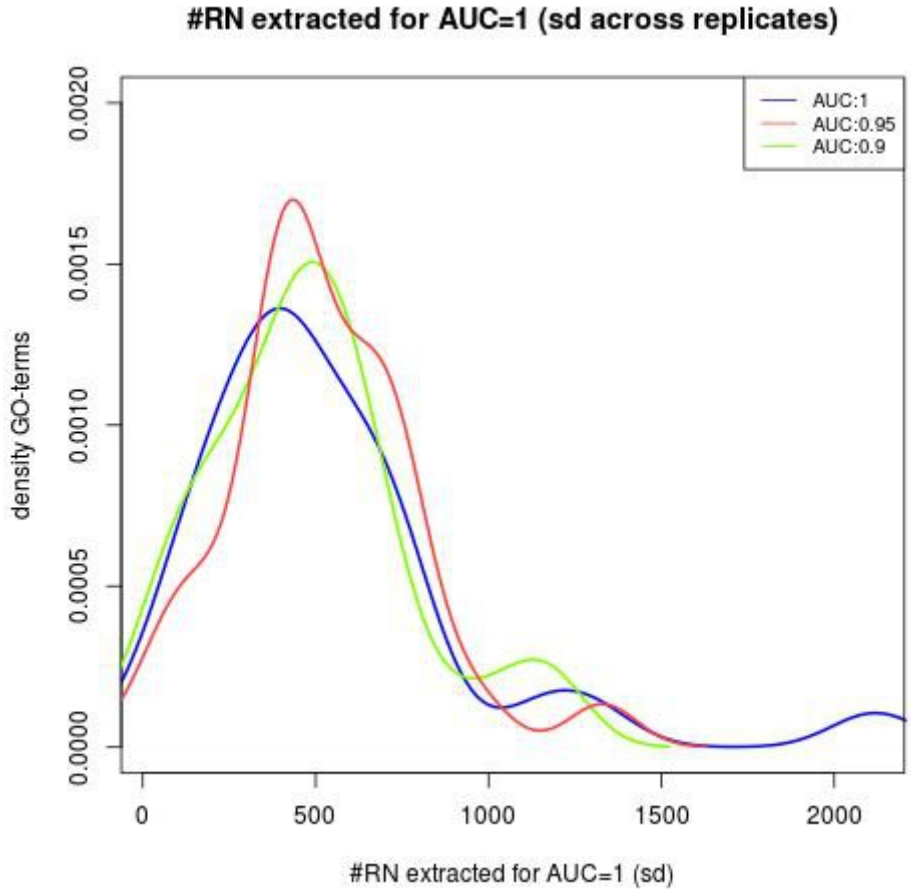
Part 4- Prediction sin chickens with BMRF

		mean sd across replicates	
#RN	BMRF	BMRF random extact	BMRF-PU
1000		0.1436	0.129
2000		0.14	0.1268
3000		0.15	0.128
4000		0.135	0.121
5000		0.13	0.127
6000		0.129	0.1244
7000		0.123	0.116
8000		0.137	0.123
all	0.033		

Var1	Var2	corr	p_value
AUC_increase	epp_V/tpEppV	0.251	0.2058
AUC_increase	te/tp_e	0.222	0.2662
AUC_increase	epp/tpEpp	0.18	0.3689
AUC_increase	epp	0.153	0.4468
AUC_increase	#genes	0.148	0.4608
AUC_increase	eppV	0.134	0.5038
AUC_increase	#genesV	0.126	0.5319
AUC_increase	spec	0.126	0.5319
AUC_increase	teV	0.124	0.5361
AUC_increase	te	0.101	0.617
AUC_increase	depth	0.085	0.673
AUC_increase	teV/tpeV	0.04	0.8425

- **Extraction of RN**


#RN extracted for AUC=1 (mean of all replicates and folds)
AUC:1
AUC:0.95
AUC:0.9
density GO-terms
0.00000
0.00010
0.00020
0.00030
2000
4000
6000
8000
10000
#RN extracted for AUC=1 (mean)


#RN extracted for AUC=1 (sd across replicates)
AUC:1
AUC:0.95
AUC:0.9
density GO-terms
0.0000
0.0005
0.0010
0.0015
0.0020
0
500
1000
1500
2000
#RN extracted for AUC=1 (sd)

Results with k:2

#RN	AUC PU.BMRF	sd AUC PU.BMRF	AUC BMRF	sd AUC BMRF
2000	0.677	0.033	0.642	0.032
4000	**0.677**	**0.034**	**0.63**	**0.032**
6000	0.655	0.033	0.624	0.028
8000	0.641	0.033	0.617	0.03

Table 10: Comparison accuracy of prediction BMRF vs PU-BMRF

Accuracy in the extraction of RN:

Average value of tolerance (sd)			
tol_AUC85	**tol_AUC90**	**tol_AUC95**	**tol_AUC1**
0.885(0.088)	0.96(0.15)	1.2(0.346)	1.42(0.438)

Table 11: Values of "tolerance" that were required for diffrenet values of AUC in the process of extraction.

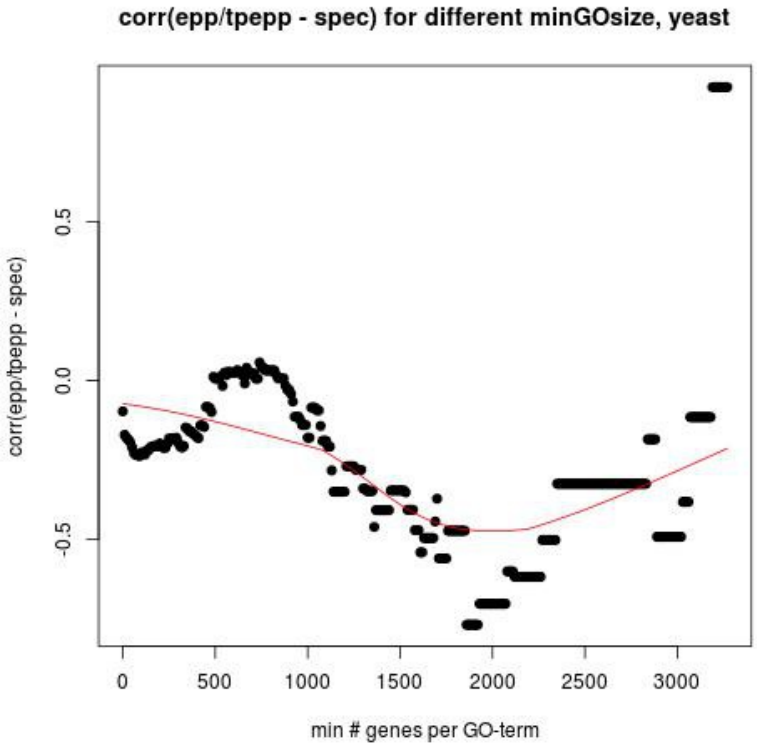
- **Computational time**

Step	Description	computational time in minutes for 1 GO-terms (% of total time)
1	Similarity Matrix	40 (18.2%)
2	Creating the folds	10 (4.5%)
3	Network features	40 (18.2%)
4	non-GO-specific features	20 (9.1%)
5	GO-specific features	60 (27.27%)
6	extraction of RN	50 (22.73%)

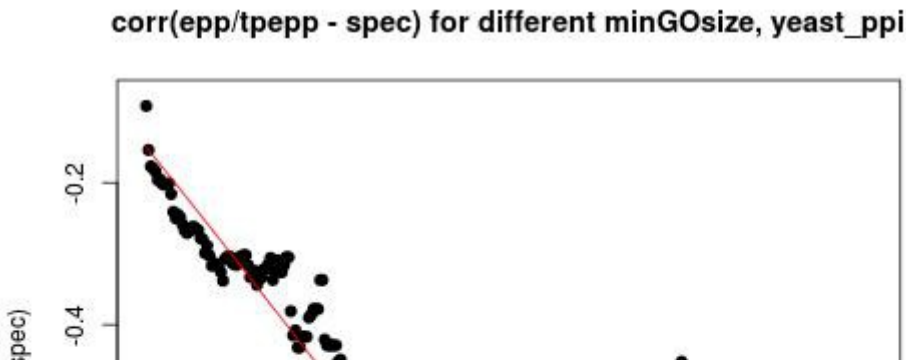
Table 12: Approximate computational time for steps of PU-BMRF for k:10 and 4 replicates

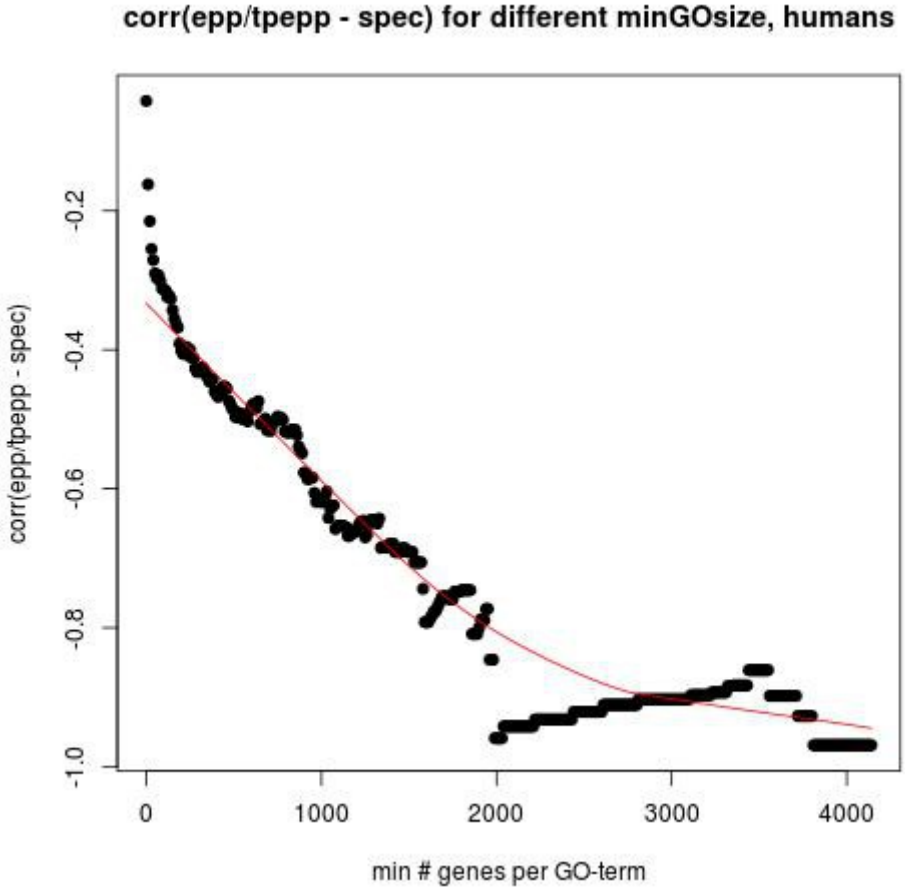
Computtaional time of BMRF for k:10 and 4 replicates is ~5 minutes. The time only increases because more analyses have to be done

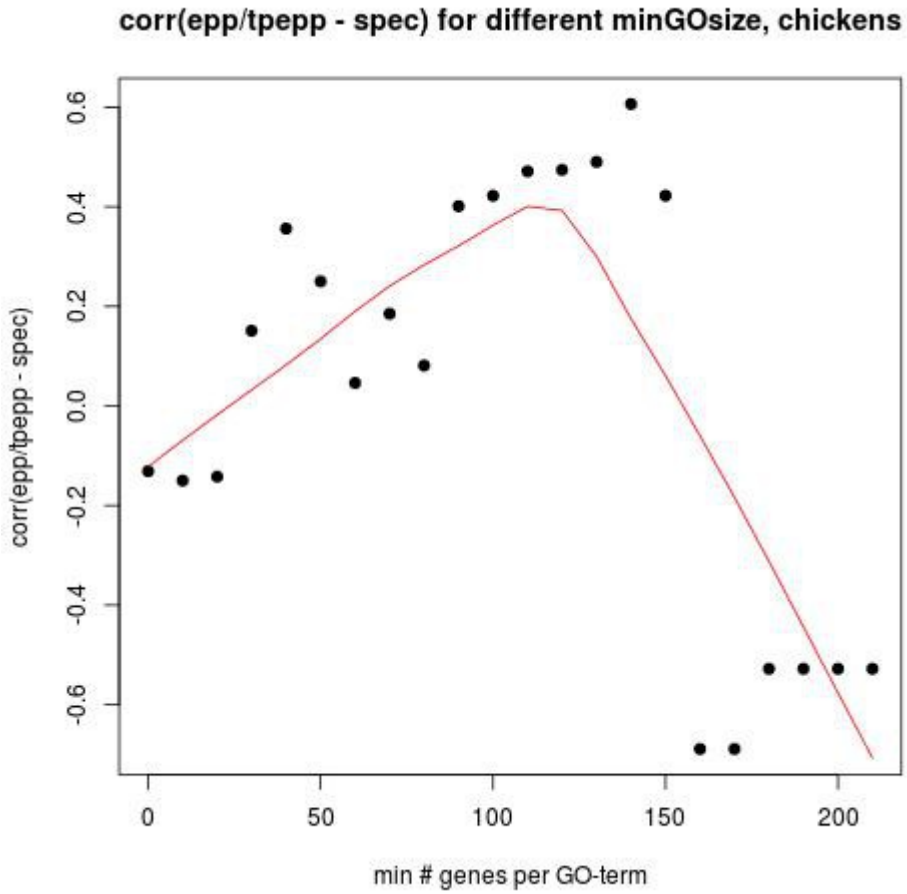
W observed a correlation of 0.67 between the number of labels of the GO-term and th eincrease in accuracy when PU-bMRF was used instead of BMRF, indicating that PU is more effective when the number of known association sis large.



Biology


corr(epp/tpepp - spec) for different minGOsize, yeast_ppi
-0.2
-0.4


corr(epp/tpepp - spec) for different minGOsize, humans
corr(epp/tpepp - spec)
-0.2
-0.4
-0.6
-0.8
-1.0
0
1000
2000
3000
4000
min # genes per GO-term



In the 3 plots the maximum correlation was when the minGOsize was around 2000.