Development of computational methods for PFP based on network data is a challenging problem in poorly annotated species. Here, we expand upon an existing BMRF and develop a PU implementation (PU-BMRF) that is more accurate than predecessor for PFP in poorly annotated species such as chicken. The efficiency of BMRF to infer function of proteins in poorly annotated had been previously noted [5,6]. Nevertheless, the efficiency of BMRF is hampered because the algorithm attempts to solve a two class classification problem when in fact the annotated data is from one single class (positive class). PU-BMRF tackles this problem by adding a previous step to the BMRF algorithm, in which a set of reliable negative are extracted. Subsequently, the BMRF classifier can be trained with a representative set of genes of each class (negative and positives) and prediction become more accurate.

The basis of PU is in extracting the genes within the set of unlabeled, that show strong differences with respect to the set of positives. It is a must that the features that are used to investigate these differences are as unrelated as possible to the features that the classifier, BMRF in this case, uses afterward. BMRF uses neighborhood information but neglects other important features like, for instance, whether the gene is associated with a related GO term, or the degree of connection between the neighbors of the gene and the genes that are associated with the GO-term of interest. We developed a PU implementation in which 64 features were taken into account to identify a set of reliable negatives (RN) for each GO term.

Lack of reprodubility

Overall, 76% of the GO-terms were more accurately predicted when PU-BMRF was used with respect to when only BMRF was used. PU-BMRF, however, have some limitations. We observed that the extraction of RN has low reproducibility. To our knowledge this lack of consistency in the extraction of RN can be explained by the low number of folds that were used in the cross validation (only two folds). In fact, probably the main drawback of PU-BMRF is that the computational time is around 200 time larger than for BMRF and this may force the user to choose a low value of k-folds in the validation. The increase in computational time was, to some extend, expected given that PU-BMRF requires the computation for 64 features for each gene within each GO term; and some of which are complex like, for instance, the sum of the closeness of a set of gene in the network. As a consequence, one may have to choose a low value of k, at the expenses of a lack of reproducibility in the extraction of RN.

The lack of reproducibility, had been previously noted in PU methods. [13], for instance extracted 20,099 and 4066 RN with two different approaches, respectively and only 589 of the RN were common between the two approaches. [7] explained that in some cases the information from GO alone is not enough to predict a good set of negative examples, because some proteins defy the conventional annotation patterns.

In line with this, it should be considered that the main goal of PU applied to PFP is to eventually predict the function of genes that do not fall in the set of positives or reliable negatives. These predictions, however, cannot be evaluated with the same method that was used to do the predictions because in most PU approaches, the genes that do not fall in the set of positives or reliable negatives are extracted form the analysis. It could be, thus, argued that the lack of reproducibility is advantageous in that the predictions can be evaluated for a larger number of genes. Since in PU-BMRF, the set of RN that are extracted differs slightly from run to run, we would expect that given a sufficient number of runs, most of the genes will be included in the set of RN at least once. Thus, due to the low reproducibility, we may be able to evaluate the predictions for nearly every gene in the database.

Both, the reproducibility and the number of genes for which the predictions are made, can to some extend, be regulated in PU-BMRF by specifying the number of RN that we want to extract in each run. The user can choose to extract a very large number of RN, although it would come at the cost of a lower increase in accuracy. In [1], for instance, they set a value of 1 as cutoff in equation XX that regulates how many RN will be extracted, whereas in [2] they chose 1.05. In our case, we changed this value according to the GO term as we aimed to extract a fixed number of RN. We decided that this was a reasonable option since we observed that the accuracy did not increase when the number of RN was very large or very small. Another common threshold to separate RN from the unlabeled data is that specificity is equal to one. Thus, the cutoff in equation XX or in any other equation that aims to separate RN form the unlabeled data, could be be defined in such a way that non of the known positives would fall in the set of RN.

Factors that determine the prediction perfromance

Overall, since PU approaches consist on adding previous step to the classification algorithms, one may say that several PU approaches can be considered (either combining or applying one after another), in order to extract a more reliable set of RN. [2], for instance, extracted a set of likely negatives (LN) based on the approached

introduced by [1] and then they extracted a set of RN using the approach developed in [3]. [13], for instance extracted RN based in two sets of features, one that consider each features individually and one that considered only the mean of the features of each group of features, and then extracted those RN that were extracted with both approaches. An important aspect to consider is that the quality of the method will not depend so much on the number of features that are defined or on how many PU approaches are integrated, but rather on two main factors: (1) How different the Positives are from the negatives for the features considered (data properties and quality of the features) and (2) Which portion of the unlabeled genes will be extracted as RN. Regarding to the fist factor, [7] referred to the so-called "moonlighting" proteins to those proteins that having a unique combination of features cannot be predicted from the conventional annotations.

In addition to this, there is another important factor that determined not only the accuracy with which the RN are extracted, but also the accuracy with which the final predictions are made via the final classifier. This factor is the extend to which the principle of gilt by association holds. It was previously shown that this "guilt by association" heuristic is universal and preserved beyond organism boundaries [12, 13]. However, the same greatly depends on the quality of the data. In particular, in the case of the co-expression networks, it should be considered that even if the quality of the data is good, the phenotypic variation is controlled at many levels, some of which are independent of transcript abundance.

Avenues of improvemnet

The current method can integrate protein-protein-interaction (ppi) data with the co-expression data, as well as data from some well-annotated related species, like, *Mus musculus* or *Homo sapiens* in the case of chicken, and this could lead to and improvement ion accuracy of prediction as explained in [5,6]. The main aspect to be improved regarding PFP in poorly annotated species like chicken, however, is not the accuracy of prediction but the number of GO-terms for which the predictions can be made. This number is very low even when the GO-size filters were at its minimum (minGOsize:9, 321 GO-terms). This problem, in-fact, may be even more sever in PU-BMRF than in BMRF, because since the computational time is much larger for PU-BMRF, the value of k may be lower, and subsequently a larger number of positives cases is required.

The number of GO terms for which make predictions can be made, could be improved, for instance, by expanding upon the existing R function "glmnet" that BMRF uses to train the classifier, in such a ways that the matrix are less sparse or that sparser matrix can be solved. In fact, probably the best avenue of improvement for PU-BMRF is be to extend the "glmnet" function to more than two classes. This would allow to do predictions within the set of unlabeled genes, taking simultaneously into account the information from the set pf positives and the set of negatives. More importantly, the matrix would become less sparse and predictions could be made for a lower number of positives.

In order to reduce the computational time of PU-BMRF, a first step would be to identify and discard the features that are weakly contributing to extraction of RN. Also, to discard those features whose values change depending on which genes are in the training set. Were these features discarded, the computation of features would need to be computed only once. This could lead to a significant decrease of the running time given that the computation of features accounts for nearly 82% of the running time of PU-BMRF.

Further aspects to be improved are, for instance, a more accurate extraction of RN, for instance, using Self organizing maps as explained in [13], or the Rochio technique [1]; accounting for the magnitude of coexpression; or including an option to make predictions for individual genes rather than for each GO-term.

[1]7.BHARDWAJ, N., GERSTEIN, M. & LU, H. 2010. Genome-wide sequence-based prediction of peripheral proteins using a novel semi-supervised learning technique. Bmc Bioinformatics, 11.
[2] http://www.mdpi.com/1420-3049/22/9/1463/htm#B16-molecules-22-01463
[3]10.JIANG, M. & CAO, J. Z. 2016. Positive-Unlabeled Learning for Pupylation Sites Prediction. Biomed Research International.
[13]https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-017-1546-7
5.JOACHIM W. BARGSTENA, EDOUARD I. SEVERINGB, J.-P. N., GABINO F. SANCHEZ-PEREZA & DIJK, A. D. J. V. 2014. Biological process annotation of proteins across the plant kingdom. Current Plant Biology, 1.
6.KOURMPETIS, Y. A. I., VAN DIJK, A. D. J., BINK, M., VAN HAM, R. & TER BRAAK, C. J. F. 2010. Bayesian Markov Random Field Analysis for Protein Function Prediction Based on Network Data. Plos One, 5.
7. http://cs.nyu.edu/media/publications/youngs_noah.pdf