Validating module network learning algorithms using simulated data

T. Michoel*♮[1], S. Maere♮[1], E. Bonnet[1], A. Joshi[1], Y. Saeys[1], T. Van den Bulcke[2], K. van Leemput[3],
P. van Remortel[3], M. Kuiper[1], K. Marchal[2,4] and Y. Van de Peer[1]

[1] Bioinformatics & Evolutionary Genomics, Plant Systems Biology, VIB, Ghent University, Technologiepark 927, B-9052 Ghent, Belgium
[2] ESAT-SCD, K.U.Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium
[3] ISLab, Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, B-2020 Antwerpen, Belgium
[4] CMPG, Department Microbial and Molecular Systems, K.U.Leuven, Kasteelpark Arenberg 20, B-3001 Heverlee, Belgium

* Corresponding author ♮ Contributed equally

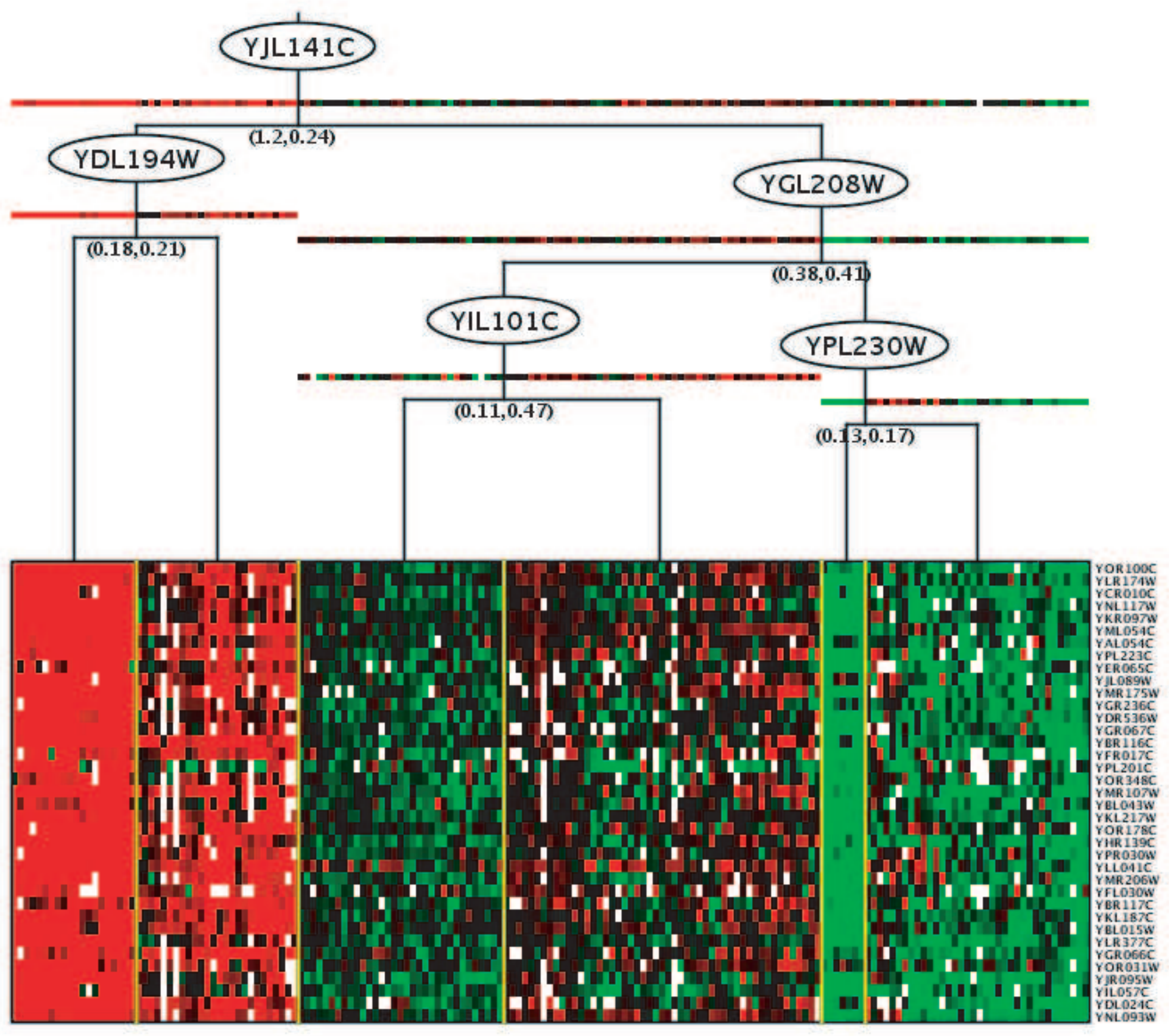

Figure 1: Sample module learned from the Gasch dataset [3]. Red and green hues indicate upregulation resp. downregulation. The pairs (x, y) under each split in the regulation tree represent the Bayesian score gain over the split, normalized on the number of genes in the complete network (x), and the regulator assignment entropy (y).

Background

In recent years, several authors have used probabilistic graphical models to learn expression modules and their regulatory programs from gene expression data (Figure 1). Despite the demonstrated success of such algorithms in uncovering biologically relevant regulatory relations, further developments in the area are hampered by a lack of tools to compare the performance of alternative module network learning strategies. Here, we demonstrate the use of the synthetic data generator SynTReN [1] for the purpose of testing and comparing module network learning algorithms. We introduce a software package for learning module networks, called LeMoNe, which incorporates a novel strategy for learning regulatory programs. Novelties include the use of a bottom-up Bayesian hierarchical clustering to construct the regulatory programs, and the use of a conditional entropy measure to assign regulators to the regulation program nodes. Using SynTReN data, we test the performance of LeMoNe in a completely controlled situation and assess the effect of the methodological changes we made with respect to an existing software package, namely Genomica [2]. Additionally, we assess the effect of various parameters, such as the size of the dataset and the amount of noise, on the inference performance.

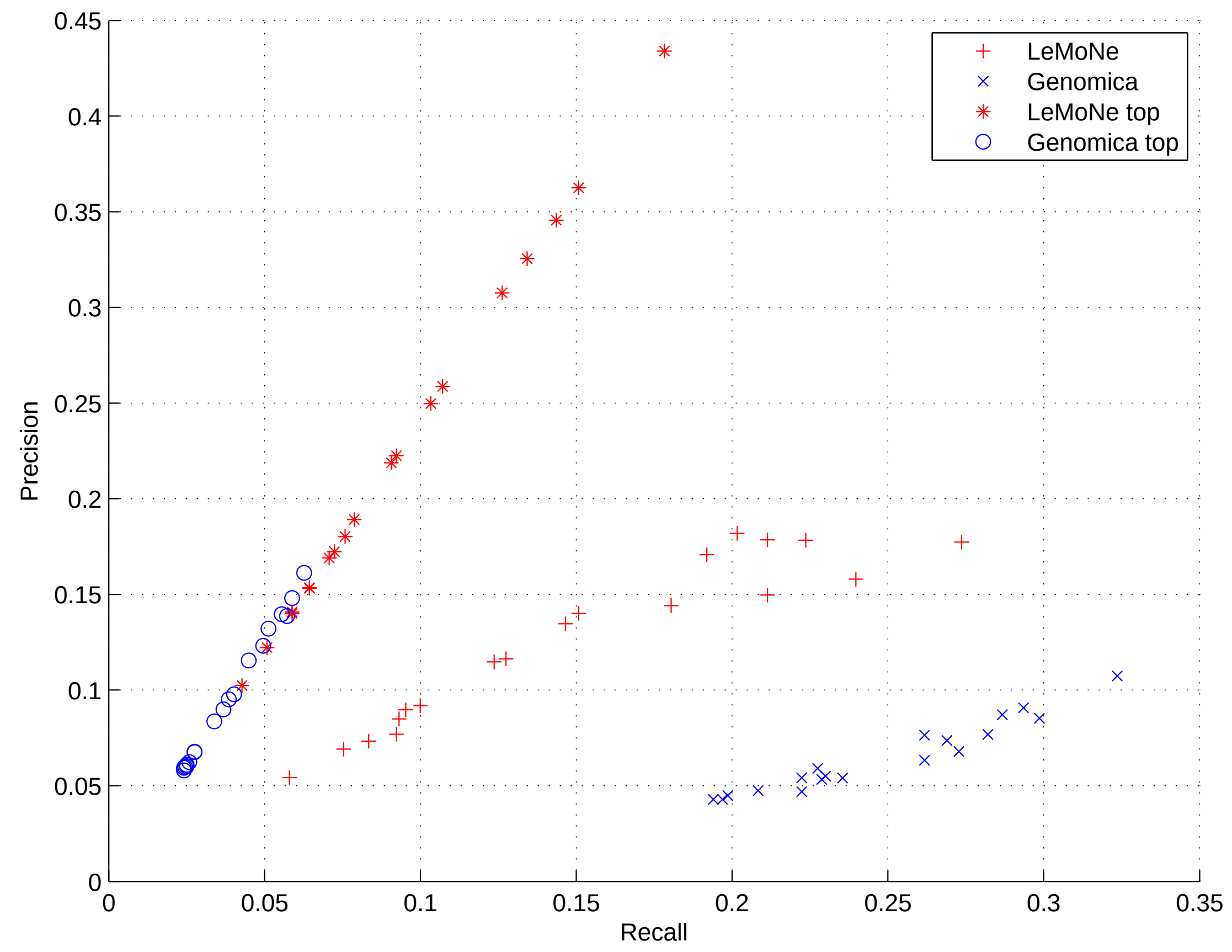

Figure 2: Comparison of heuristic search methods by recall – precision pairs for data sets with 1000 genes, 100 experiments and different noise levels, for the complete output network, and for the subnetwork generated by the top regulators in the regulation programs.

Results

Overall, application of Genomica and LeMoNe to simulated datasets gave comparable results (Figure 2). However, LeMoNe offers some advantages:

- The learning process is considerably faster for larger datasets.
- The location of the regulators in the LeMoNe regulation programs (Figure 2, 3) and their conditional entropy (Figure 4) may be used to prioritize regulators for functional validation.
- The combination of the bottom-up clustering strategy with the conditional entropy-based assignment of regulators improves the handling of missing or hidden regulators.

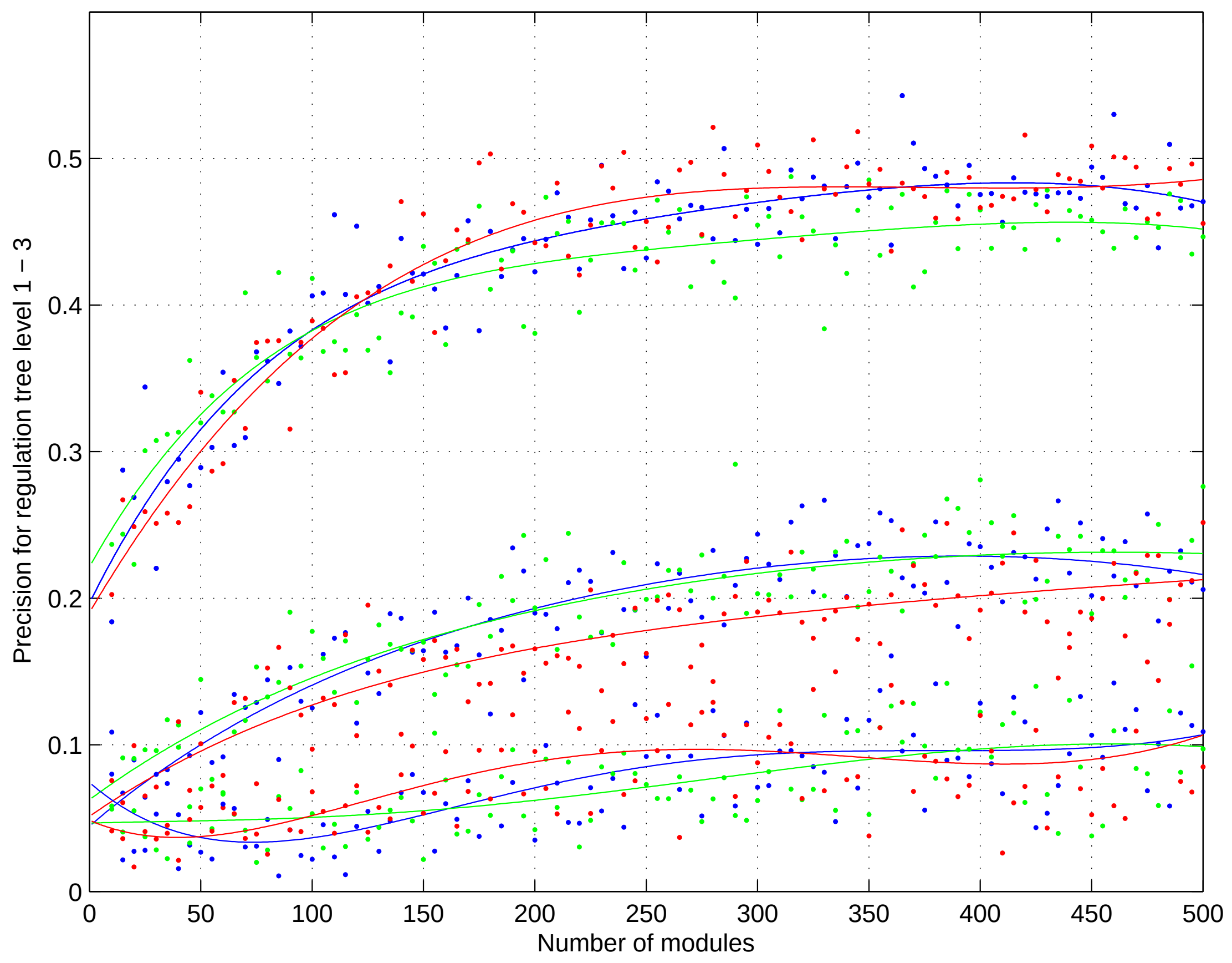

Figure 3: Precision as a function of the number of modules for subnetworks generated by regulation tree levels 1 (roots), 2 and 3 (top to bottom) for data sets with 1000 genes and 100 (red), 200 (green) and 300 (blue) experiments. The curves are least squares fits of the data to a linear non-polynomial model of the form $a_0 + \sum_{k=1}^{n} a_k x^{k-1} e^{-x/500}$ with x the number of modules and $n = 3$.

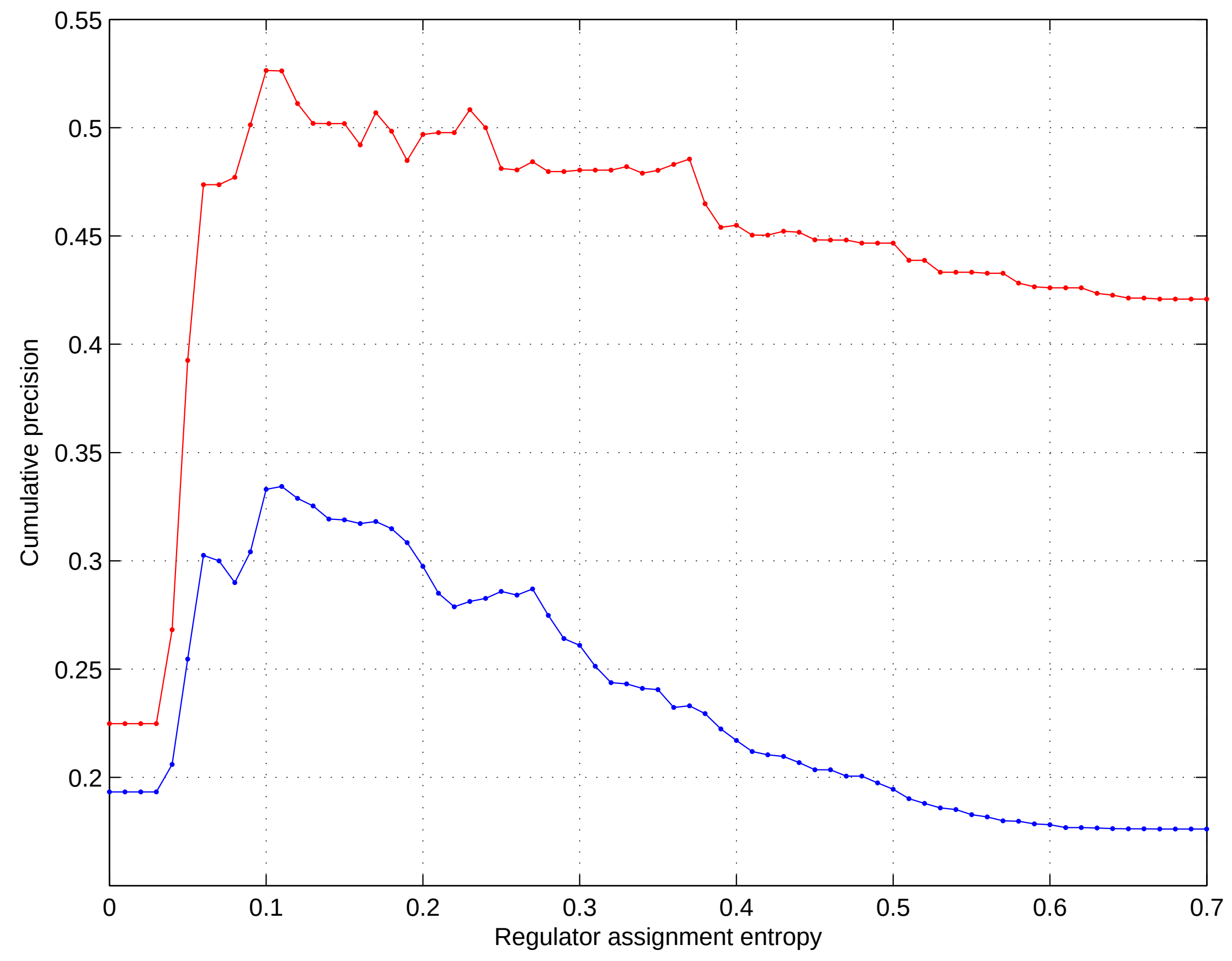

Figure 4: Cumulative precision as a function of regulator entropy for a data set with 1000 genes and 100 experiments, with 150 modules: each point at an entropy value x (spaced at 0.01 intervals) gives the precision of all (blue) or top (red) regulators with assignment entropy $\leq x$.

Conclusions

We show that data simulators such as SynTReN are very well suited for the purpose of developing, testing and improving module network algorithms. We used SynTReN data to develop and test an alternative module network learning strategy, which is incorporated in the software package LeMoNe, and we provide evidence that this alternative strategy has several advantages with respect to existing methods. We expect that further development of our results, such as introducing more advanced elements of information theory into the learning process, will further improve the performance of module network learning algorithms.

References

[1] Van den Bulcke T, *et al.*: *BMC Bioinformatics* 2006, **7**:43.

[2] Segal E, *et al.*: *Nat Genet* 2003, **34**:166 – 167.

[3] Gasch AP, *et al.*: *Mol Biol Cell* 2000, **11**:4241 – 4257.

Paper and software

Michoel T, Maere S, *et al.*, **Validating module network learning algorithms using simulated data**, *BMC Bioinformatics*, *in press* 2007.

`<http://bioinformatics.psb.ugent.be/LeMoNe/download.htm>`