SciForum
Mol2Net

Pairwise Ortholog Detection in Related Yeast Species by Using Big Data Supervised Classifications

Deborah Galpert Cañizares [1], Sara del Río García [2], Francisco Herrera [2], Evys Ancede Gallardo [3], Agostinho Antunes [4,5], Guillermin Agüero-Chapin [4,*]

[1] Departamento de Ciencias de la Computación, Universidad Central ¨Marta Abreu¨ de Las Villas (UCLV), Santa Clara, 54830, Cuba; E-Mail: deborah@uclv.edu.cu

[2] Dept. of Computer Science and Artificial Intelligence, CITIC-UGR, University of Granada, Granada, Spain; E-Mails: srio@decsai.ugr.es (S.R.G.); herrera@decsai.ugr.es (F.F.)

[3] Centro de Bioactivos Químicos, Universidad Central ¨Marta Abreu¨ de Las Villas (UCLV), Santa Clara, 54830, Cuba; E-Mail: eancedeg@uclv.edu.cu

[4] CIMAR/CIIMAR, Centro Interdisciplinar de Investigação Marinha e Ambiental, Universidade do Porto, Rua dos Bragas, 177, 4050-123 Porto, Portugal; E-Mail: aantunes@ciimar.up.pt

[5] Departamento de Biologia, Faculdade de Ciências, Universidade do Porto, Rua do Campo Alegre, 4169-007 Porto, Portugal

* Author to whom correspondence should be addressed; E-Mail: gchapin@ciimar.up.pt.

Received: 1 October 2015 / Accepted: 1 October 2015 / Published: 2 December 2015

Abstract: Orthology detection still requires more effective scaling algorithms. Combinations of alignment, synteny, evolutionary distances and protein interactions have been used in different unsupervised algorithms to improve effectiveness while many available databases are concerned with the scaling problem. In this paper, a set of gene pair features based on similarity measures, such as alignment scores, sequence length, gene membership to conserved regions and physicochemical profiles are combined in a supervised Pairwise Ortholog Detection (POD) approach to improve effectiveness considering low ortholog ratios in relation to all possible pairwise comparisons between two genomes. In this POD scenario, big data supervised classifiers managing imbalance between ortholog and non-ortholog pair classes allow for an effective scaling solution built from two genomes and extended to other genome pairs. The supervised approach for POD was compared with Reciprocal Best Hits (RBH), Reciprocal Smallest Distance (RSD) and a Comprehensive, Automated Project for the Identification of Orthologs from Complete Genome Data (OMA) algorithms by using (i) *Saccharomyces cerevisiae - Kluyveromcyes lactis*, (ii) *Saccharomyces cerevisiae - Candida glabrata* and (iii) *Saccharomyces cerevisiae - Schizosaccharomyces pombe* yeast genome pairs as benchmark datasets. Four datasets derived from each genome pair comparison with different alignment settings were used. Because of the large amount of instances (gene pairs) and the data imbalance, the building and testing of the supervised model was only possible by using big data supervised classifiers managing imbalance. Evaluation metrics taking low ortholog ratios into account were applied. From the effectiveness perspective, MapReduce Random Oversampling combined with Spark Support Vector Machines outperformed RBH, RSD and OMA, probably, because of the consideration of gene pair features beyond alignment similarities combined with the advances in big data supervised classification.

Keywords: ortholog detection; big data supervised classification; similarity measures

1. Introduction

Ortholog detection (OD) algorithms should distinguish orthologous genes from other types of homologs such as paralogs evolving from a common ancestor through a duplication event. A great deal of unsupervised graph-based approaches has been developed to identify orthologs resulting in corresponding repositories for pre-computed orthology relationships.

When OD is based only on sequence similarity, it has been limited by evolutionary processes such as recent paralogy events, horizontal gene transfers, gene fusions and fissions, domain recombinations or different genetic events [1-2]. In fact, the identification of homologs is a difficult task in the presence of short sequences, those that evolved in a convergent way, and the ones that share less than 30% of amino acid identities (twilight zone). Algorithm failures have been particularly shown in benchmark datasets from *Saccharomycete* yeast species that underwent whole genome duplications (WGD) presenting rampant paralogies and differential gene losses [3]. To tackle these shortcomings, some OD solutions merge sequence similarity with synteny genome rearrangements, protein interactions, domain architectures and evolutionary distances.

On the other hand, the integration of different gene or protein information and the massive increase in complete proteomes highly increase the dimensionality of the OD problem and the total number of proteins to be classified. In a thorough paper from the Quest for Orthologs consortium [4], the authors emphasize the idea that this increase in proteome data brings out the need to work out not only efficient but effective OD algorithms. As they mention, the increase in computational demands in sequence analyses is not easily met by an increase in computational capacities but rather calls for new approaches or algorithmic implementations [4]. They summarized some methodological shortcuts implemented by the existing orthology databases to deal with the scaling problem.

In this paper, we propose a new supervised approach for pairwise OD (POD) that combines several gene pairwise features (alignment-based and synteny measures with others derived from the pairwise comparison of the physicochemical properties of amino acids) to address big data problems [4]. Our big data supervised POD approach allows scaling to related species and data imbalance management (low ortholog ratio found in two or more genomes) for an effective OD. The methodology consists of three steps: (1) the calculation of gene pair features to be combined, (2) the building of the classification model using machine learning algorithms to deal with big data from a pairwise dataset, and (3) the classification of related gene pairs.

Since traditional supervised classifiers cannot scale large datasets, the supervised classification for the POD problem should be addressed as a big data classification problem according to [5-7] and big data solutions should be applied for binary classification in imbalanced data such as the ones presented in [8].

Finally, we evaluate the application of several big data supervised techniques that manage imbalanced datasets [8-9] such as cost-sensitive Random Forest (RF-BDCS), Random Oversampling with Random Forest (ROS+RF-BD) and the Apache Spark Support Vector Machines (SVM-BD) [9] combined with MapReduce ROS (ROS+SVM-BD). The effectiveness of the supervised approach is compared to RBH, RSD and OMA algorithms, taking data imbalance into account. All the algorithms were evaluated on benchmark datasets derived from the following yeast genome pairs: *S. cerevisiae* and *K. lactis, S. cerevisiae* and *C. glabrata* [3] and *S. cerevisiae* and *S. pombe* [10]. The *S. cerevisiae* and *C. glabrata* pair is particularly complex for OD since both species had undergone WGD. We found that our supervised approach outperformed traditional methods, mainly when we applied ROS combined with SVM-BD.

2. Results and Discussion

For the evaluation of POD algorithms, we compare the supervised solutions and the unsupervised ones following the evaluation scheme in Figure 1. The process separates the pairs into train and test sets and calculates pairwise similarity measures (average of local and global alignment similarity measures, length of sequences, gene membership to conserved regions (synteny), and physicochemical profiles within 3, 5 and 7 window sizes) for the pairs of both sets. The sequences of the test sets should be used to run the unsupervised reference algorithms. The train set should be used for building the supervised models to be tested only with the test set.

The performance quality evaluation involves the calculation of the Geometric Mean (*G-Mean*) [11], seeking to maximize the accuracy of the two classes (orthologs and non-orthologs) by achieving a good balance between sensitivity and specificity that consider misclassification costs; and the Under the ROC Curve (*AUC*) [12] to show the classifier performance over a range of data distributions [13].

In Experiment 1, we evaluated the algorithms inside a genome by partitioning at random 75% of the complete set of pairs for training and 25% for testing, while in Experiment 2 we built the model from a genome pair and tested it in two different pairs. Specifically, in Experiment 1 we divided the *S. cerevisiae* - *K. lactis* set into 16.986.996 pairs for training and 5.662.332 pairs for testing. The four datasets (BLOSUM50, BLOSUM62_1, BLOSUM 62_2 and PAM250) of each genome pair were built from combinations of alignment parameter settings. On the other hand, in Experiment 2, we built the classification model from 22.649.328 pairs of *S. cerevisiae* and *K. lactis* genomes and tested it in 29.887.416 pairs of *S. cerevisiae* and *C. glabrata*, and 8.095.907 pairs of *S. cerevisiae* and *S. pombe* genomes.

Comparison of big data supervised classifiers

The *G-Mean* values of the supervised algorithms change only slightly with the selection of different alignment parameters (Table 1). These results may be either caused by the aggregation of global and local alignment scores in a single similarity measure or by the appropriate combination of scoring matrices and gap penalties in relation to the sequence diversity between the two yeast genomes [14].

The average results of *AUC* and *G-Mean* obtained in experiments 1 and 2 for the supervised algorithms with different parameter values are shown in Table 1. The average TP_{Rate} and TN_{Rate} are also depicted in Figure 2. SVM-BD has been left out from the table due to its very poor performance in *G-Mean* caused by its imbalance between TP_{Rate} and TN_{Rate}. Both Table 2 and Figure 2 prove that big data supervised classifiers managing imbalance outdo their corresponding big data supervised versions.

The ROS pre-processing method for big data makes SVM-BD useful for POD and improves the performance of RF-BD even more with a higher value for the resampling size parameter of 130% [15]. In contrast, both experiments show that the variation in this parameter value from 100% to 130% does not significantly influence on the performance of the SVM-BD classifier with different regulation values.

Specifically, RF-BDCS shows the best performance in *S. cerevisiae* - *C. glabrata* and *S . cerevisiae* - *K. lactis* when the classification quality is measured by *G-Mean* and *AUC* metrics, because it enhances the learning of the minority class. The criterion used to select the best tree split is based on the weighting of the instances according to their misclassification costs, and such costs are also considered to calculate the class associated with a leaf [8]. This cost treatment does not explicitly change the sample distribution and avoids the possible overtraining, that it is present in the ROS solutions due to replicated cases. The election of the cost values ($C(+|-) = IR$ and $C(-|+) = 1$) may also define the success of the algorithm.

In the case of SVM-BD, the fixed regularization parameter defines the trade-off between the goal of minimizing the training error (i.e., the loss) and minimizing the model complexity to avoid overfitting. The higher is its value, the simpler the model. Nonetheless, setting an intermediate value, or one close to cero may produce a better performance in classification [16]. This is the case of the ROS (RS: 100%) + SVM-BD (regParam: 0.5) classifier that exhibits the best *AUC* and *G-Mean* values in *S. cerevisiae* - *S. pombe*, and the best balance between TP_{Rate} and TN_{Rate} in the three datasets (Figure 2).

In order to balance time with classification quality, time consumption is another aspect to have in mind when comparing big data solutions. Table 3 contains run time in seconds for all big data solutions in each dataset and the faster algorithms are highlighted in bold face. These results allow us to prove that the time required is directly related to the operations needed for each method, as well as to the size of the datasets used to build the model. The fastest algorithm considering the average run time is SVM-BD followed by SVM-BD combined with ROS. Thus, the fastest algorithms coincide with the ones with better performance. In general, the ROS (RS: 100%) + SVM-BD (regParam: 0.5) classifier can be considered the best supervised solution considering both performance and time.

Comparison of supervised vs. unsupervised classifiers

The average results of *AUC* and *G-Mean* obtained for the best supervised algorithms and the unsupervised algorithms with different parameter values are shown in Table 4 for experiments 1 and 2. The supervised classifiers outperform the unsupervised ones. Among the unsupervised algorithms, RSD reaches the highest *G-Measure* value by setting E-value = 1e-05 and α = 0.8 (recommended values in [17]) in *S. cerevisiae* - *C. glabrata* where similar results can also be seen for *AUC* and TP_{Rate} values. On the contrary, OMA was the best among the unsupervised algorithms in *S. Cerevisiae* - *S. pombe* datasets (Table 4).

In general, the performance of all classifiers declined in *S. Cerevisiae* - *S. pombe* datasets due to the fact that *S. pombe* is a distant relative of *S. cerevisiae* [18]. The supervised classifiers performance is affected for the same reason and also, by the difference in data distribution between the train and test sets [19]. On the contrary, ROS (RS: 100%) + SVM-BD (regParam: 0.5) remained stable in *S. Cerevisiae*

- *C. glabrata* and *S. Cerevisiae* - *S. pombe* datasets when considering the balance between TP_{Rate} and TN_{Rate} . Superior results in *S. cerevisiae* - *C. glabrata* are outstanding, since both genomes underwent a WGD and a subsequent differential loss of gene duplicates, so that algorithms are prone to produce false positives. Thus, this dataset contains "traps" for OD algorithms [3].

The reduced quality shown by RBH, RSD and OMA, mainly in the case of RBH, could be caused by their initial assumption that the sequences of orthologous genes/proteins are more similar to each other than they are to any other genes from the compared organisms. This assumption may produce classification errors [1], in spite of the fact that BLAST parameters can be tuned as has been recommended in [20]. Conversely, RSD not only compares the sequence similarity, but it relies on maximum likelihood estimation of evolutionary distances to detect orthologs between two genomes, and as a result, it finds many putative orthologs missed by RBH because it is less likely than RBH to be misled by existing close paralogs.

The OMA algorithm also displays advantages over RBH. It uses evolutionary distances instead of alignment scores. This algorithm allows the inclusion of one-to-many and many-to-many orthologs. It also considers the uncertainty in distance estimations and detects potential differential gene losses.

From the point of view of the intrinsic information managed by the algorithms, the success of big data supervised classifiers managing imbalance over RSD and OMA may be explained by feature combinations calculated for the datasets together with the learning from curated classifications. With the aggregation of global and local alignment scores we are combining protein structural and functional relationships between sequence pairs, respectively. Besides, we incorporate other gene pair features: (i) the periodicity of the physicochemical properties of amino acids that allows us to detect similarity among protein pairs in their spectral dimension [21]; (ii) the conserved neighbourhood information, which considers that genes belonging to the same conserved segment in genomes of different species will probably be orthologs; and (iii) the length of sequences

Table 1. Geometric mean results of the best supervised classifiers in each dataset.

Dataset	**ROS (RS: 100%) + RF-BD (Scer-Klac)**	**ROS (RS: 130%) + RF-BD (Scer-Klac)**	**RF-BDCS (Scer-Klac)**	**ROS (RS: 100%) + RF-BD (Scer-Cgla)**	**ROS (RS: 130%) + RF-BD (Scer-Cgla)**	**RF-BDCS (Scer-Cgla)**	**ROS (RS: 100%) + SVM-BD (regParam: 1.0) (Scer-Spombe)**	**ROS (RS: 100%) + SVM-BD (regParam: 0.5) (Scer-Spombe)**
Blosum50	0.9818	0.9818	**0.9896**	0.9889	0.9885	**0.9934**	0.8393	**0.8673**
Blosum621	0.9801	0.9818	**0.9855**	0.9891	0.9903	**0.9932**	0.8707	**0.8959**
Blosum622	0.9793	0.9793	**0.9905**	0.9910	0.9910	**0.9929**	0.8536	**0.8694**
Pam250	0.9818	0.9818	**0.9899**	0.9912	0.9905	**0.9941**	0.8495	**0.8839**

Table 2. *AUC* and *G-Mean* results of supervised classifiers in experiments 1 and 2.

	S.cerevisiae-S.Klactis		*S.cerevisiae-C.glabrata*		*S.cerevisiae-S.pombe*	
Algorithm	***AUC***	***G-Mean***	***AUC***	***G-Mean***	***AUC***	***G-Mean***
RF-BD	0.6979	0.6291	0.7455	0.7005	0.5172	0.1851
ROS (RS: 100%)+RF-BD	0.9809	0.9807	0.9901	0.9900	0.6096	0.4527
ROS (RS: 130%)+RF-BD	0.9813	0.9812	0.9901	0.9901	0.6121	0.4581
RF-BDCS	**0.9889**	**0.9889**	**0.9934**	**0.9934**	0.7294	0.6745
ROS (RS: 100%) + SVM-BD (regParam: 1.0)	0.9477	0.9477	0.9542	0.9542	0.8632	0.8533
ROS (RS: 100%) + SVM-BD (regParam: 0.5)	0.8845	0.8791	0.9540	0.9539	**0.8845**	**0.8791**
ROS (RS: 100%) + SVM-BD (regParam: 0.0)	0.6135	0.4961	0.9432	0.9431	0.6135	0.4961
ROS (RS: 130%) + SVM-BD (regParam: 1.0)	0.8164	0.7956	0.9523	0.9522	0.8164	0.7956
ROS (RS: 130%) + SVM-BD (regParam: 0.5)	0.8629	0.8528	0.9539	0.9539	0.8629	0.8528
ROS (RS: 130%) + SVM-BD (regParam: 0.0)	0.6248	0.5147	0.9429	0.9428	0.6248	0.5147

Table 3. Run time results in seconds of the big data solutions in experiments 1 and 2.

Algorithm	*S.cerevisiae-S.Klactis*	*S.cerevisiae-C.glabrata*	*S.cerevisiae-S.pombe*
RF-BD	1201.59	2174.90	2060.99
ROS (RS: 100%)+RF-BD	2983.75	4562.38	4440.03
ROS (RS: 130%)+RF-BD	3345.04	4805.50	4681.51
RF-BDCS	1302.41	2362.04	2025.15
SVM-BD	**461.87**	**482.85**	**480.45**
ROS (RS: 100%) + SVM-BD (regParam: 1.0)	**867.38**	**1011.59**	**1012.46**
ROS (RS: 100%) + SVM-BD (regParam: 0.5)	**874.62**	**1008.77**	**1013.32**
ROS (RS: 100%) + SVM-BD (regParam: 0.0)	**859.17**	**1008.24**	**999.31**
ROS (RS: 130%) + SVM-BD (regParam: 1.0)	927.14	1079.19	1079.58
ROS (RS: 130%) + SVM-BD (regParam: 0.5)	929.17	1084.19	1076.33
ROS (RS: 130%) + SVM-BD (regParam: 0.0)	924.42	1076.37	1077.21

Table 4. *AUC* and *G-Mean* of the unsupervised and the best supervised classifiers.

	S. cerevisiae-.K. lactis		*S. cerevisiae-C .glabrata*		*S. cerevisiae-S. pombe*	
Algorithm	***AUC***	***G-Mean***	***AUC***	***G-Mean***	***AUC***	***G-Mean***
RBH	0.1497	0.0062	0.8196	0.7995	0.4697	0.4525
RSD 0.2 1e-20	0.5862	0.4862	0.9238	0.9206	0.4874	0.4438
RSD 0.5 1e-10	0.5926	0.4643	0.9340	0.9316	0.4980	0.4063
RSD 0.8 1e-05	0.5886	0.4518	0.9382	0.9362	0.5009	0.3899
OMA	0.5765	0.4904	0.9287	0.9259	0.5151	0.4644
RF-BDCS	**0.9889**	**0.9889**	**0.9934**	**0.9934**	0.7294	0.6745
ROS (RS: 100%) + SVM-BD (regParam: 1.0)	0.9477	0.9477	0.9542	0.9542	0.8632	0.8533
ROS (RS: 100%) + SVM-BD (regParam: 0.5)	0.8845	0.8791	0.9540	0.9539	**0.8845**	**0.8791**

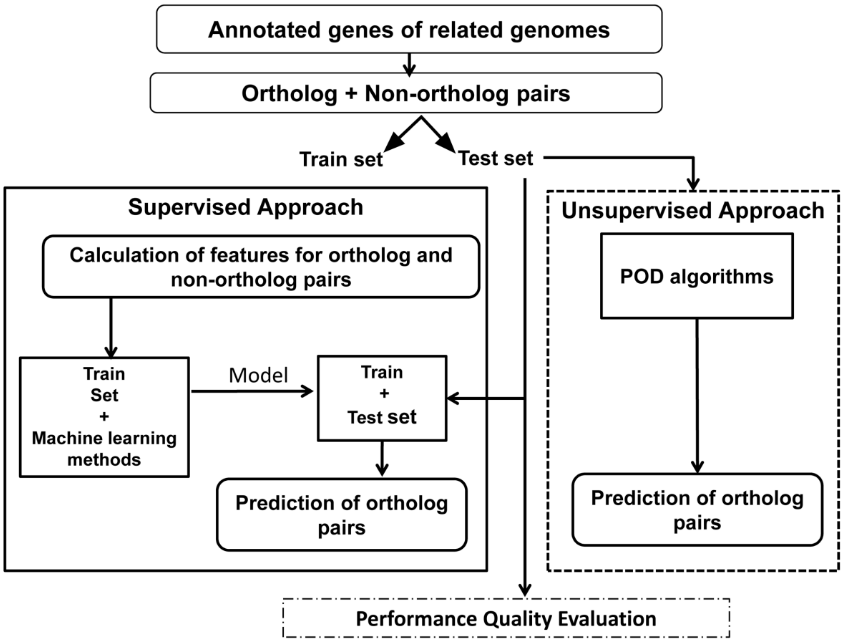
Annotated genes of related genomes
Ortholog + Non-ortholog pairs
Train set
Test set
Supervised Approach
Calculation of features for ortholog and non-ortholog pairs
Train Set + Machine learning methods
Model
Train + Test set
Prediction of ortholog pairs
Unsupervised Approach
POD algorithms
Prediction of ortholog pairs
Performance Quality Evaluation

Figure 1. Workflow of the evaluation of supervised *vs*. unsupervised POD algorithms.

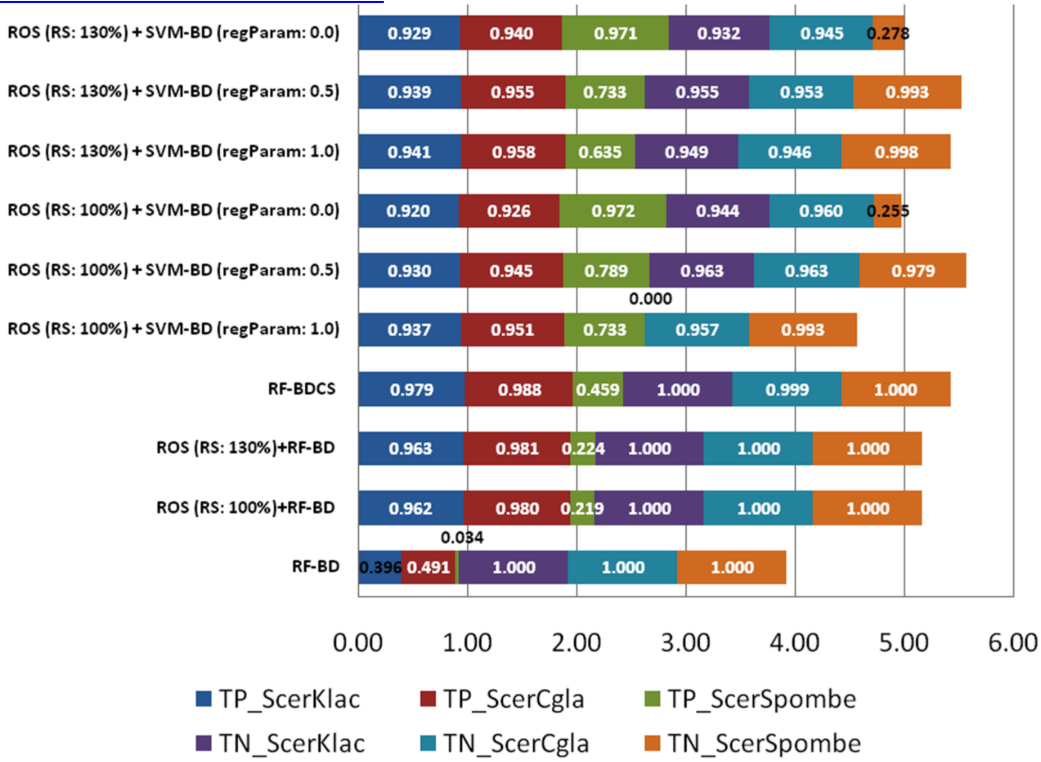

Figure 2. Average true positive and true negative rate values of supervised classifiers obtained in experiments 1 and 2.

3. Materials and Methods

Datasets

The characteristics of the datasets are summarized in Table 5 where the label #Atts represents the number of attributes or gene pair features, and #Class (maj; min), the number of pairs in both classes. *S. cerevisiae* - *S. pombe* dataset contains ortholog pairs representing 95.18% of the union of the Inparanoid7.0 and GeneDB classifications described in [10]. On the other hand, *S. cerevisiae* - *K. lactis* and *S. cerevisiae* - *C. glabrata* datasets contain all ortholog pairs in the gold groups reported in [3]. When we built the set of instances with all possible pairs, we excluded some genes since we didn't find their genome physical location data in the YGOB database [22], required for the conserved membership feature calculation.

Big data supervised classification managing data imbalance

We use the open-source project Hadoop [23] with its highly scalable and fault-tolerant Hadoop Distributed File System (HDFS). We also utilize the scalable Mahout data mining and machine learning library [24] with machine learning algorithms adapted according to the MapReduce scheme as the MapReduce implementation of the (Random Forest (RF) algorithm [25]. Finally, we use the Apache Spark framework [9] interacting with HDFS, when the implementation of SVM-BD in the scalable MLLib machine learning library [16] is combined with the MapReduce ROS implementation [8].

Table 5. Characteristics of the datasets.

Genome pair	#Atts	#Class (maj; min)	Imbalance ratio (*IR*)	Excluded genes
S. cerevisiae - K. lactis 1	6	(22.646.914; 2414)	9381.489	89 de 5861 genes de *S. cerevisiae* 37 de 5215 genes de *C. glabrata* 1403 de 5327 genes de *K. lactis*
S. cerevisiae - C. glabrata 1	6	(29.884.575; 2841)	10519.034	
S. cerevisiae - S. pombe 2	6	(8.090.950; 4.957)	1632.227	

4. Conclusions

The development of effective supervised algorithms for POD in a big data scenario was made possible by: (i) the availability of curated databases (authentic orthologs), (ii) the combination of traditional alignment measures with other gene pair features (sequence length, gene membership to conserved regions and physicochemical profiles) to complement homology detection, and (iii) the treatment of the low ratio of orthologs to the total possible gene pairs between two genomes. By applying evaluation metrics such as *G-mean*, AUC and the balance between TP_{Rate} and TN_{Rate}, our results show that gene pairwise feature combinations provide excellent POD in a big data supervised scenario that consider data imbalance. The SVM-BD classifier combined with the ROS (RS: 100%) pre-processing with regulation parameter 0.5 outdid the rest of the big data supervised solutions and the popular unsupervised (RBH, RSD and OMA) algorithms even when the supervised model was extended to datasets containing "traps" for OD algorithms. The classification performance of the supervised algorithms measured by *G-Mean* and AUC metrics did not significantly change in the four test sets obtained with different alignment parameter settings. When the balance between time and classification quality is considered, ROS (RS: 100%) + SVM-BD (regParam: 0.5) also proves to be the algorithm of choice. In future research, the introduction of new gene pair features might improve the effectiveness and efficiency of the supervised algorithms for POD.

Acknowledgments

GACh acknowledge the Portuguese Fundação para a Ciência e a Tecnologia (FCT) for financial support with reference (SFRH/BPD/92978/2013). AA was partially supported by the European Regional Development Fund (ERDF) through the COMPETE - Operational Competitiveness Programme and national funds through FCT under the projects PEst-C/MAR/LA0015/2013 and PTDC/AAC-AMB/121301/2010 (FCOMP-01-0124-FEDER-019490). This work was also partially supported by the Spanish Ministry of Science and Technology under project TIN2014-57251-P and the Regional Andalusian Research Projects P11-TIC-7765 and P10-TIC-6858.

Author Contributions

Conceived and designed the experiments: DGC and GACh. Performed the experiments: DGC, SRG and EAG. Analyzed the data: DGC, SRG, FH and GACh, Contributed reagents/materials/analysis tools: FH, EAG, and AA. Wrote the paper: DGC, SRG and GACh. Critically revised the manuscript: GACh, FH and AA.

Conflicts of Interest

The authors declare no conflict of interest.

References and Notes

1. Kristensen, D.M.; Wolf, Y.I.; Mushegian, A.R.; Koonin, E.V. Computational methods for gene orthology inference. *Briefings in bioinformatics* **2011**, *12*, 379-391.
2. Kuzniar, A.; Ham, R.C.H.J.v.; Pongor, S.; Leunissen, J.A.M. The quest for orthologs: Finding the corresponding gene across genomes. *Trends in Genetics* **2008**, *30*, 1-13.

3. Salichos, L.; Rokas, A. Evaluating ortholog prediction algorithms in a yeast model clade. *PLoS ONE* **2011**, *6*, 1-11.
4. Sonnhammer, E.L.L.; Gabaldón, T.; Sousa da Silva, A.W.; Martin, M.; Robinson-Rechavi, M.; Boeckmann, B.; Thomas, P.D.; Dessimoz, C.; Orthologs, c.Q.f. Big data and other challenges in the quest for orthologs. *Bioinformatics Editorial* **2014**, 1-6.
5. Fernández, A.; Río, S.d.; López, V.; Bawakid, A.; Jesus, M.J.d.; Benítez, J.M.; Herrera, F. Big data with cloud computing: An insight on the computing environment, mapreduce, and programming frameworks. In *WIREs Data Mining Knowl Discov*, 2014.
6. Beyer, M.; Laney, D. 3d data management: Controlling data volume, velocity and variety. http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf
7. Chen, C.L.P.; Zhang, C.Y. Data-intensive applications, challenges, techniques and technologies: A survey on big data. *Information Sciences* **2014**, *275*, 314–347.
8. Río, S.d.; López, V.; Benítez, J.M.; Herrera, F. On the use of mapreduce for imbalanced big data using random forest. *Information Sciences* **2014**, *285*, p. 112-137.
9. Zaharia, M.; Chowdhury, M.; Das, T.; Dave, A.; Ma, J.; McCauley, M.; Franklin, M.; Shenker, S.; Stoica, I. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *9th USENIX Conference on Networked Systems Design and Implementation*, San Jose, CA, 2012; pp 1-14.
10. Koch, E.N.; Costanzo, M.; Bellay, J.; Deshpande, R.; Chatfield-Reed, K.; Chua, G.; D'Urso, G.; Andrews, B.J.; Boone, C.; Myers, C.L. Conserved rules govern genetic interaction degree across species. *Genome Biology* **2012**, *13*.
11. Barandela, R.; Sánchez, J.S.; García, V.; Rangel, E. Strategies for learning in class imbalance problems. *Pattern Recognit.* **2003**, *36*, 849–851.
12. Bradley, A.P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition* **1997**, *30*, 1145–1159.
13. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering* **2009**, *21*, 1263-1284.
14. Pearson, W.R. Selecting the right similarity-scoring matrix. *Current Protocols in Bioinformatics* **2013**, *43*, 3.5.1-3.5.9.
15. Triguero, I.; Río, S.d.; López, V.; Bacardit, J.; Benítez, J.M.; Herrera, F. Rosefw-rf: The winner algorithm for the ecbdl'14 big data competition: An extremely imbalanced big data bioinformatics problema. *Knowledge-Based Systems* **2015**.
16. Krishnan, S.; Smith, V. Linear support vector machines (svms). <https://spark.apache.org/docs/latest/mllib-linear-methods.html#linear-support-vector-machines-svms> (January 2015),
17. DeLuca, T.F.; Wu, I.-H.; Pu, J.; Monaghan, T.; Peshkin, L.; Singh, S.; Wall, D.P. Roundup: A multi-genome repository of orthologs and evolutionary distance. *Bioinformatics* **2006**, *22*, 2044-2046.
18. Wood, V.; Piskur, P.J. Schizosaccharomyces pombe comparative genomics; from sequence to systems. In *Topics in Current Genetics*, P. Sunnerhagen, J. Piškur (Eds.): Comparative Genomics ed.; Springer-Verlag Berlin Heidelberg 2005: 2005; Vol. 15

19. Moreno-Torres, J.G.; Llorà, X.; Goldberg, D.E.; Bhargava, R. Repairing fractures between data using genetic programming-based feature extraction: A case study in cancer diagnosis. . *Information Sciences* **2013**, 805-823.
20. Hagelsieb, G.M.; Latimer, K. Choosing blast options for better detection of orthologs as reciprocal best hits. *Bioinformatics* **2008**, *24*, 319-324.
21. Carpio-Muñoz, C.A.D.; Carbajal, J.C. Folding pattern recognition in proteins using spectral analysis methods. *Genome Informatics* **2002**, *13*, 163-172
22. Byrne, K.P.; Wolfe, K.H. The yeast gene order browser: Combining curated homology and syntenic context reveals gene fate in polyploid species. *Genome Research* **2005**, *15*, 1456–1461.
23. White, T. Hadoop, the definitive guide.
24. Owen, S.; Anil, R.; Dunning, T.; Friedman, E. Mahout in action. Manning Publications Co.: 2011.
25. Hakim, D.A. Partial data mapreduce random forests. https://mahout.apache.org/users/classification/partial-implementation.html