ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

E. Weitschek, R. Van Velzen, G. Felici

SPECIES CLASSIFICATION USING DNA BARCODE SEQUENCES: A COMPARATIVE ANALYSIS

R. 11-07 2011

E. Weitschek – Institute of Systems Analysis and Computer Science "A. Ruberti", National Research Council, Viale Manzoni 30, 00185 Rome, Italy and Department of Informatics and Automation, Università degli studi "Roma Tre", Via della Vasca Navale 79, 00146 Rome, Italy - emanuel.weitschek@iasi.cnr.it..

R. Van Velzen – Biosystematics Group, Wageningen University, Wageningen, The Netherlands..

G. Felici – Institute of Systems Analysis and Computer Science "A. Ruberti", National Research Council, Viale Manzoni 30, 00185 Rome, Italy..

ISSN: 1128–3378

Collana dei Rapporti
Istituto di Analisi dei Sistemi ed Informatica “Antonio Ruberti”
Consiglio Nazionale delle Ricerche

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.cnr.it
URL: http://www.iasi.cnr.it

Abstract

Background
With the term Barcode we refer to a short fragment of mitochondrial DNA composed of few hundreds of bases from which it appears to be possible to extract the information needed to classify living species. Such intuition has been confirmed by many experimental results and different data analysis methods have been proposed that tackle this problem within the standard paradigm of Machine Learning. In this work we consider 3 different methods that are available to tell the species of an organism from the analysis of its Barcode, and compare them over a large number of real and simulated data.

Results
We describe and compare three DNA Barcode data analysis methods, Neighbor Joining (NJ), Nearest Neighbor (NN) and Barcoding with LOGic Formulas (BLOG). For the comparison we use simulated as well as published empirical data sets. Neighbor Joining is a tree based method and Nearest Neighbor is a lazy learning distance based method. BLOG is a logic data mining character based data analysis method; in the experiments we use a new improved version of this logic data mining software - BLOG 2.0. The improvements in the methodology of this tool and in its software design lead to higher accuracy and recognition rates over a large test bed of empirical and simulated data sets.

Conclusions
The performed experiments show that precise and effective species classification using DNA barcodes can be achieved with the three methods. In particular, we show that on average BLOG outperforms Neighbor Joining and Nearest Neighbor both in descriptive and predictive power, supplying also compact models in terms of logic formulas, which are able to correctly classify large numbers of different specimens.

1. Background

Specimen classification with DNA Barcode sequences was originally proposed by Hebert et al. in [1]. This technique is based on a short DNA sequence taken from a small portion of the mitochondrial DNA, the gene Cytochrome C Oxidase subunit I (COI), to be used as a species "Barcode", that differs by several percent, even in closely related species, and collects enough information to identify the species of an individual. Several data analysis methods have been developed and adopted to automatically classify a DNA Barcode sequence in a predefined species. In this study, we present three DNA Barcode data analysis methods: Neighbour Joining, Best Match and BLOG 2.0. Neighbour Joining [2] is the most used method in DNA Barcode data analysis. It is a bottom-up clustering method used for the construction of phylogenetic trees based on sequence distance. Best Match [3] is a distance based method, which is actually considered the best performing Barcode data analysis method [4]. BLOG 2.0 is an evolution of the logic data mining method described in [5]. It is able to identify small characterizing nucleotide positions of DNA Barcode sequences and to classify the different species with logic formulas. We compare the success rates of the BLOG method with those of distance-based methods (Neighbour Joining and Best Match) in terms of descriptive and predictive power by testing the accuracy with simulated DNA Barcode datasets.

The aim of this study is to compare three DNA Barcode classification methods and to indicate which method performs best in terms of classification rates in order to guide the scientists in the selection of the optimal method for DNA Barcode data analysis.

2. Methods

2.1 DNA Barcode data simulation

Realistic DNA Barcode datasets were simulated using Mesquite version 2.72 build 528 [6]. We simulated along two axes: time of species divergence and effective population size. First a random ultrametric species tree (S) for 50 species was simulated using the Yule model [7], [8], with a total tree length of 1 million generations. Then, ultrametric gene trees (T) were simulated on that species tree S according to a coalescence process with 20 specimens per species. Gene trees were simulated using effective population sizes of 1,000 10,000 and 50,000 individuals, using 100 replicates in each category of population size. Non-ultrametric gene trees (G) were then obtained by adding noise to branch lengths of the ultrametric gene trees (T). The noise was generated according to a normal distribution with variance of 0.7 times the original branch length.
DNA sequences were simulated on the non-ultrametric gene trees (G) according to a HKY model of sequence evolution which was selected as the best fitting model for a representative empirical dataset of 527 Nymphalidae DNA Barcodes by JModelTest 0.1.1 [9]. Model parameters encompassed a transition/transversion ratio of 8.3, nucleotide frequencies of 0.30 (A), 0.15 (C), 0.10 (G), 0.45 (T), and gamma-distributed rate variation over sites with 4 site categories and a shape parameter of 0.2. A sequence length of 650 base pairs was adopted, approximating the length of the standard DNA Barcode for animals (cytochrome oxidase I). Simulated datasets were converted to fasta format using the seqCleaner perl script [10] before analysis, and the specimens were divided over a train set with 15 specimens per species and a test set with 5 specimens per species. The train set was considered as DNA Barcode reference libraries containing specimens of a priori known species membership; the specimens in the test sets were considered unknown DNA Barcodes queries.
In the following sections we describe the methods adopted for the comparative study. These methods were compared based on their ability to describe the sequences in the train sets and to correctly identify query sequences in the test sets.

2.2 BLOG: Barcoding with LOGic formulas

BLOG is a logic data mining method based on two main algorithmic steps:
(i) the selection of the most relevant DNA base pairs that are best candidates to distinguish the different species;
(ii) the extraction of the logic rules that are able to identify precisely a species(formula extraction).
The method is described in the next paragraphs.

2.2.1 Data representation

Each specimen in the data is assigned to one and only one species (according to the format already described in [4]). The data set is composed of *n* specimens, belonging to two or more species; we

refer to the attributes of the specimen as features. The i^{th} object of the data set is represented by the vector $f_i = (f_{i1}, f_{i2},..., f_{im})$, where f_{ij} is a numeric integer value that maps the base of the j^{th} feature; the data matrix is represented by the sequence of vectors $f_1, f_2,..., f_n$. Given this matrix representation of the data set, when appropriate the specimens may also be referred as rows, while the features as columns. To simplify the description, we assume that the specimens belong to one of only two species, species A and species B.

2.2.2 Feature selection

Feature selection (FS) is the extraction of a small subset of attributes that are able to characterize a data sample. In barcoding the feature selection step objective is to choose the characteristic species specific DNA bases. We adopt a formulation of the feature selection problem as a mathematical optimization problem where an objective function that measures the information retained by the selected features is to be maximized under some constraints. Such formulation is proven to be effective in many real applications [5], [11]. A complete description of several variants of this formulation is available in [8]. Similar results are presented in [12], where the authors describe a combinatorial problem that represents the feature selection problem. These formulations derive from the original *test cover problem* presented in [13].

We point out that this approach can be applied to both discrete and numerical data; for the latter case, a discretization step need to be applied (as it is the case, for example, for the applications described in [8] and [14]).

The only drawback of these formulations is that the problem size grows quadratically with the specimens in the data set and they easily become computationally expensive to solve when training data reaches significant dimensions. To overcome this drawback, a linear approximation of the formulation is proposed in [5] for DNA Barcode analysis. This variant keeps the problem resolution time in a reasonable range, while experimental evidence shows that the quality of the results is not compromised.

To any extent, if we deal with very large data sets the formulation can become intractable for state of the art solvers and we need to resort to ad-hoc heuristic solution approaches that do not provide proof of optimality of the solution. An effective heuristic algorithm for this problem is described in [15]: a heuristic and non-deterministic GRASP (Greedy Randomized Adaptive Search Procedure) algorithm. This meta-heuristic approach, initially proposed in [16], is a randomized multistart iterative heuristic.

To select the specific features that are able to distinguish a species from all the other ones, we define a distinct feature selection problem for each species that is represented in the training data; in this problem the selected species is considered as a class (say class A) while all the other species are considered as another class (say class B). Clearly this approach requires a large amount of computation as it needs to solve *m* different instances of the FS problem (where *m* is the number of species in the training set). For this reason the use of a fast and effective heuristic algorithm – as the one described in [8] - is required to solve the experiment of reasonable size, as the ones that will be described in the following sections.

2.2.3 The classification step

As described in [5] after the feature selection step the logic mining system Lsquare extracts, for each species, the separating formulas. The Lsquare method is based on a reformulation of the formula extraction step as a sequence of Minimum Cost Satisfiability Problem (*MinSat*), a well studied combinatorial problem whose solution can be computationally very demanding (see again [11] for details on *MinSat*). The formulas extracted by this method are *Disjunctive Normal Form*

(DNF) over literals associated with the presence or the absence of a base in a position of the specimen. A complete description of the Lsquare method and on its strategy to solve the *MinSat* formulation adopted can be found in [17].
A fine tuning of the formula extraction computation is done by increasing the cost of a negative literals, in order to obtain a majority of positive literals and possibly to avoid to have negated literals at all. This is done for simplifying the taxonomist's work, who can deal with more readable formulas; besides, positive literals are more specific and hence have better predictive value.
The above step produces a logic formulas – or logic rule – for each species. At this point, an additional evaluation step is performed on the training set to assign proper weights to them. Each logic formula is applied to each specimen of the training set and:

- if the formula recognizes the specimen and it is associated to the same species, the number of correct classified elements (true positives TP) of the formula is increased;
- if the formula wrongly recognizes the specimen and it is associated to a different species, the number of wrong classified elements (false positives FP) of the formula is increased;
- if the formula does not recognize the specimen but it is associated to the same species, then the number of not recognized elements (false negatives FN) of the formula is increased;
- if the formula does not recognize the specimen and it is associated to a different species, then the number of true negatives (TN) of the formula is increased.

These quantities are then normalized and the true positive rate (%TP), false positive rate (%FP) and true negative rate (%TN) are computed; we also consider the *Laplace score* (equal to (TP+1)/(TP+FP+*m*)) and the *false positive / true positive rate* (FP/TP).

The classification of the test data is done in the following way:

- if an element is recognized by only one formula then it is classified in the species associated to that formula;
- if an element is recognized by two or more formulas, then the formula with a higher *Laplace Score* is chosen;
- if the *Laplace Score* is equal, we consider the false positive value and select the formula with the lower false positive value;
- if also the FP value is equal then the element is not classified.

Additional cut offs of formulas with sub-optimal coverage are done by considering the false positive / true positive rate (FP/TP) and the true positive rate. If FP/TP is greater than 0.7 or the TP is 0 the formula is considered unreliable and therefore discarded.

2.2.4 The new BLOG 2.0 software

An improved version of the software has been developed. Figure 1 represents the flow chart of BLOG 2.0 with all the different software modules.

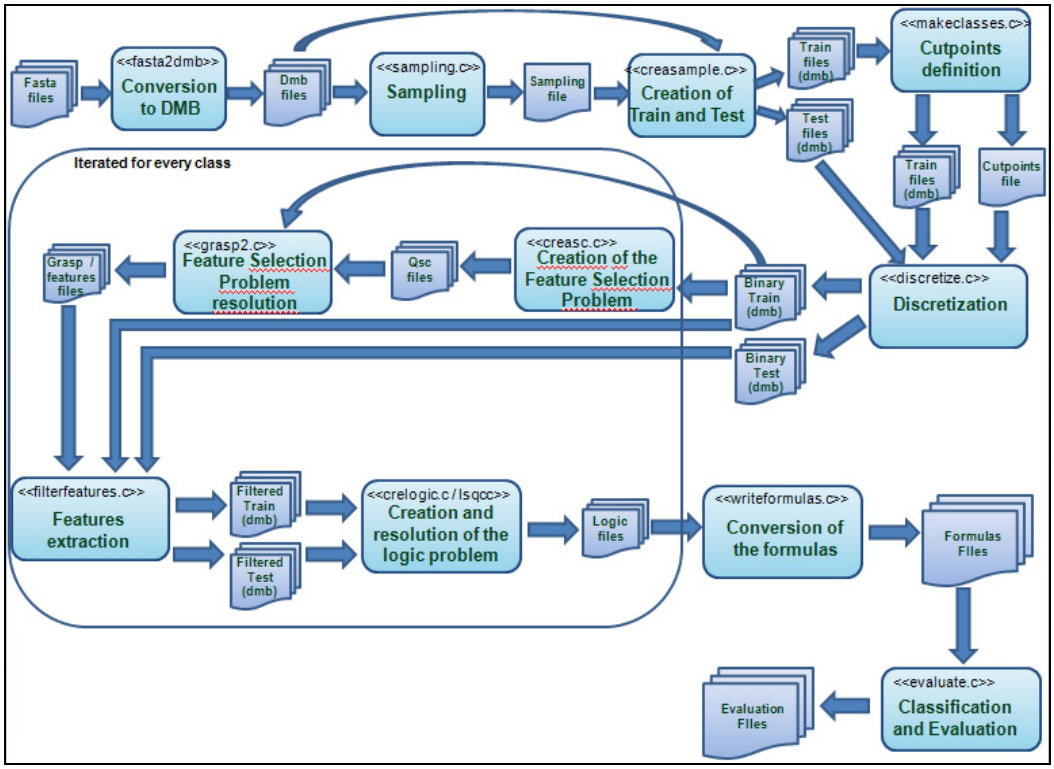

Figure 1 *The new flow chart of BLOG 2.0*

For a detailed description of the modules we point interested readers to [5]. The main differences from the previous version described in [5] is that the features selection and extraction can be iterated for each species in the data set. For computing the classification with this new flow the final part of the data mining software has been rewritten.

2.3 Neighbor Joining

The Neighbor Joining algorithm [2] is a bottom-up clustering method used for the construction of phylogenetic trees based on DNA sequence distance. Computational efficiency being its main advantage, Neighbor joining is the most widely used method for classifying DNA barcodes. The underlying assumption is that specimens of distinct species form discrete clusters in the phylogenetic tree, because genetic variation within species is smaller than that between species.
This method iteratively joins the two elements of V that are at the minimum distance. The distance is computed with the Q-matrix, an N×N matrix (where N is the number of elements) that contains in position (u, v) the distance between element u and v defined as follows:

$q_{uv} = d_{uv}(N - 2) - \sum_{w \in V} d_{uw} - \sum_{w \in V} d_{vw}$ where d_{uv} is the distance from u to v in a distance matrix D.

The neighbor joining algorithm performs the following steps:

1. compute the Q-matrix of the set of elements V from the distance matrix D;
2. add a node *w* to the tree, joining the two element ($u, v \in$ V) with the lowest value in the Q-matrix. Nodes *u* and *v* are removed from V and *w* is inserted;
3. calculate the new distances D of the nodes from the new node *w*. The new distances from *w* are defined as $d_{wz} = 0.5\ (d_{uz} + d_{vz} - d_{uv})$; and
4. start the algorithm again with the new V set and D matrix.

Sequences of unknown specimens can subsequently be included in the tree to see in which cluster they appear. We applied the neighbor joining method using the open source statistical package R [18], using the functions of the package APE 2.5–3 [19]. To calculate descriptive power we tested if conspecific specimens appear as distinct (monophyletic) clusters in trees based on the train sets. To calculate predictive power we tested if the two clusters nearest to each specimen in the test sets consist exclusively of the same species membership.

2.4 Best Match

Best match is considered to be the least stringent method for DNA barcode sequence classification. It simply assigns a query sequence to the same species membership as its closest sequence in the reference library, based on sequence distance [3]. We implemented the best match method in R [18], using the functions of the package APE 2.5–3 [19]. To test descriptive power, we tested if closest matches of all specimens in the train sets are conspecific. To test predictive power we tested if specimens in the test sets have a closest match with a train sequence of the same species membership.

3. Results and discussion

3.1 Test Plan

Comparative experiments for the above mentioned methods have been performed. First, the BLOG 2.0 software has been tested on the three data sets described in [5] and the classification results have been compared with the results obtained with the first version of BLOG. Then, the software has been examined using simulated data sets (see section 2.1) and the results evaluated according to the performances of Neighbour Joining and Best Match. In the comparative study, we use two measures of success:

- the *descriptive power*, by testing if the logic formulas produced by BLOG correctly classify all the specimens in the train sets;
- the *predictive power*, by testing if the formulas correctly classify specimens in the test sets.

3.2 BLOG 2.0 vs BLOG 1

BLOG 2.0 has been examined on three empirical data sets, two provided by the Consortium for the Barcode of Life and one obtained directly from the GenBank Nucleotide Database (http://www.ncbi.nlm.nih.gov/genbank/). The data sets are described more in detail in [1]. The experiments have been performed in the following way: we computed four runs for different dimensions of the set of selected features (10, 15, 20, 25) and with 10% and 20% random testing set. The first data set is composed of 1700 DNA barcode sequences coming from individuals belonging to 150 different species. For the first data set we report in Table 1 the average classification performances at the varying number of features:

# of Chosen features	Test %	Training % error rate	Testing % error rate
10	10	0.23	14.47
10	20	0.40	15.30
15	10	0.00	12.65
15	20	0.00	13.62
20	10	0.00	12.13
20	20	0.00	12.90
25	10	0.00	11.30
25	20	0.00	11.59
avg		**0.08**	**13.00**
std		**0.15**	**1.39**

Table 1 *Average classification performance of BLOG 2.0 for the first data set*

The error rates plot in Figure 2 restates the performances in an intuitive way:

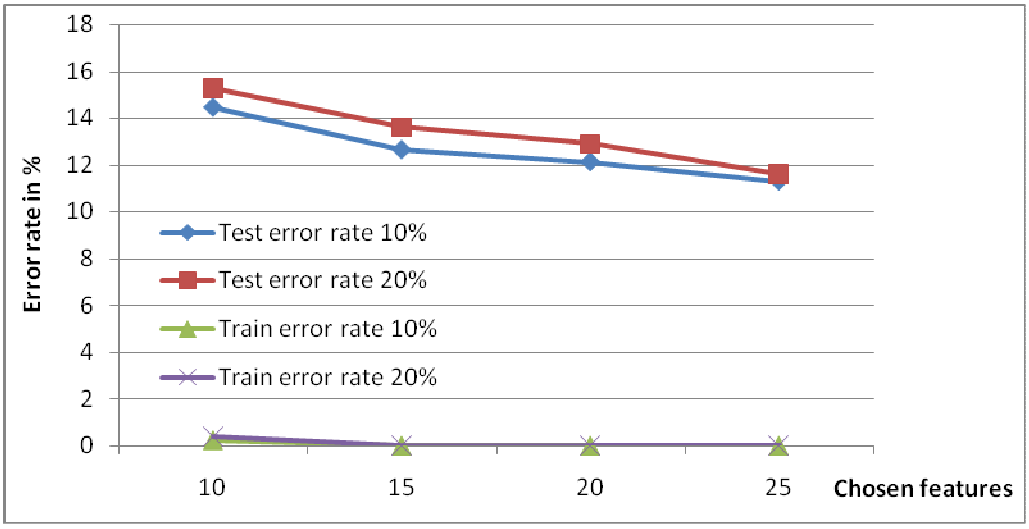

Figure 2 *Error rate plot for the first data set (BLOG 2.0)*

With respect to the first release of BLOG we note an improvement in the descriptive power (in the training set the error rate is almost 0; in many experiments we have a perfect model that fits the training set). In terms of predictive power the new version of BLOG performs slightly worse on this dataset, due to an overfitting behavior of the model.

In Table 2 we list the logic formulas for the first 5 classes. The interpretation of the formulas is straight-forward: e.g., the second line of the Table has the interpretation: “if position 70 has base G and position 82 has base T, then the specimen is classified in species A2”. All specimens belonging to species A2 in the training samples satisfy this rule (coverage = 1.00), while no specimens of different species do so (0 False Positive).

Species	*Formula*	*Coverage*	*False Negative*	*False Positive*	*Score (Laplace)*	*FP/TP*
A1	76=C AND 82=A AND 127=A AND 151=T AND 196=T AND 202=A	1.00	0	0	0.057	0
A2	70=G AND 82=T	1.00	0	0	0.026	0
A3	136=A AND 196=C AND 202=T	1.00	0	0	0.032	0
A4	127=T AND 136=A AND 250=T	1.00	0	0	0.032	0
A5	28=C AND 67!=A AND 127=A AND 154=T	1.00	0	0	0.032	0

Table 2 *Logic formulas for the first data set (BLOG 2.0)*

The second data set is composed of 826 DNA barcode sequences of 82 species of bats (Chiroptera). In Table 3 and Figure 3 we summarize the classification error rates:

# of Chosen features	Test %	Training % error rate	Testing % error rate
10	10	0.00	3.18
10	20	0.00	3.08
15	10	0.00	2.96
15	20	0.00	3.12
20	10	0.00	3.05
20	20	0.00	3.13
25	10	0.00	3.01
25	20	0.00	2.85
avg		0.00	3.05
std		0.00	0.11

Table 3 *Average classification performance of BLOG 2.0 for the second data set*

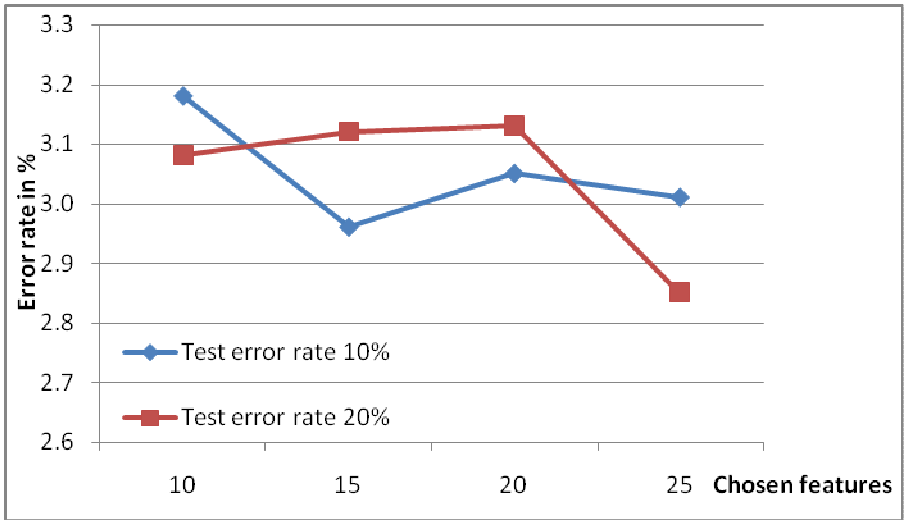

Figure 3 *Error rate plot for the second data set (BLOG 2.0)*

The logic formulas for the first 5 species are listed in Table 4.

Species	*Formula*	*Coverage*	*False Negative*	*False Postive*	*Score (Laplace)*	*FP/TP*
Ametrida centurio	182=G AND 215=C AND 554=A	1.00	0	0	0.11	0
Anoura caudifer	320=A AND 539=G	1.00	0	0	0.147	0
Anoura geoffroyi	215=G AND 320=G AND 542=A	1.00	0	0	0.214	0
Anoura latidens	56=C AND 215=A AND 554=A	1.00	0	0	0.047	0
Artibeus amplus	56=T AND 104=T AND 320=T AND 539=C	1.00	0	0	0.069	0

Table 4 *Logic formulas for the second data set (BLOG 2.0)*

With respect to the first release of BLOG with this dataset we have an improvement in the descriptive power: in all experiments we have a perfect model that fits the training set. Even in term of predictive power the new version of Blog performs better, as the average error rate decreases to 3.05% respect to 10.45%.

The third data set was obtained from GenBank Nucleotide Database and is composed of 626 Barcode sequences coming from specimens belonging to 82 different species of fish (all Craniata except Tetrapoda). In Table 5 and Figure 4 the classification error rates are reported:

# of Chosen features	Test %	Training % error rate	Testing % error rate
10	10	0.34	6.04
10	20	0.34	6.32
15	10	0.25	5.20
15	20	0.21	5.26
20	10	0.23	5.20
20	20	0.23	5.20
25	10	0.23	5.20
25	20	0.23	5.20
avg		0.26	5.45
std		0.05	0.46

Table 5 *Average classification performance of BLOG 2.0 for the third data set*

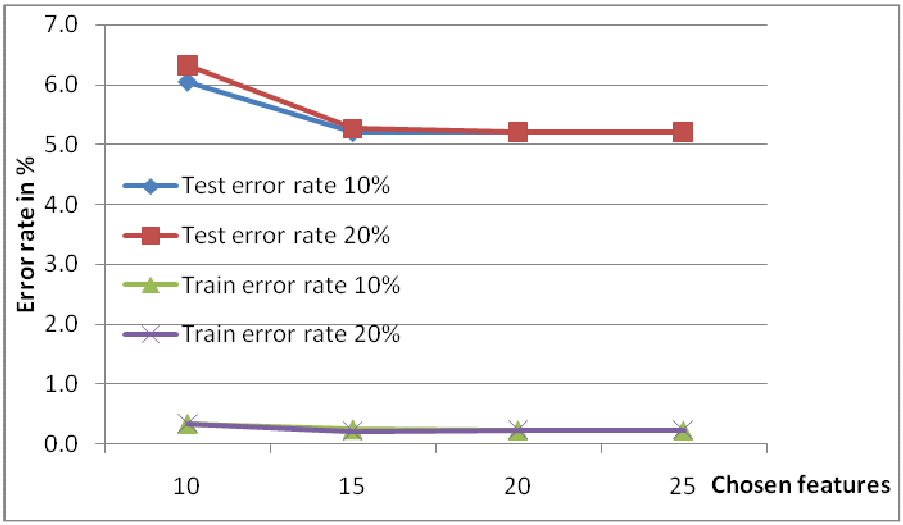

Figure 4 *Error rate plot for the third data set (BLOG 2.0)*

The new improvements of BLOG lead to an increase in the correct recognized elements rate of 4% in term of descriptive and predictive power.

The logic formulas for the first 5 species of this dataset are listed in Table 4.

Species	*Formula*	*Coverage*	*False Negative*	*False Postive*	*Score (Laplace)*	*FP/TP*
Ompok bimaculatus	282=T AND 305=A AND 382=A	1.00	0	0	0.2703	0
Ompok pabo	311=A AND 391=G AND 409=A	1.00	0	0	0.2703	0
Glyptothorax ventrolineatus	258=A AND 364=G AND 409=A	1.00	0	0	0.0899	0
Glyptothorax brevipinnis	23=A AND 209=G AND 320=G OR 23=G AND 317=G AND 329=G	1.00	0	0	0.173	0
Parambassis ranga	57=T AND 117=A AND 317=A AND 346=A	1.00	0	0	0.047	0

Table 6 *Logic formulas for the third data set (BLOG 2.0)*

The computational experiments show that BLOG 2.0 performs better than the previous version in term of descriptive power over all the 3 datasets, while is predictive power is improved for the second and third datasets while being slightly reduced for the first one. The formulas have also the most literals in positive form, which is an improvement in the explicative effect of the model.

3.3 BLOG 2.0 vs NJ and BM

In this section we report the results of the experiments on simulated data sets (more details on the generation method are given in section 2.1) comparing the performances of BLOG, Neighbour Joining and Best Match. 3 data sets each of 1000 individuals with different effective population size: 1000, 10000 and 50000 have been generated. We performed 100 runs for every data sets and in each run the training and testing split (80% - 20%) was generated randomly and exclusively respect to the previous generated data sets. In Table 7 we report the success rates of the three classification methods. A representation of the results is also given in Figures 5 and 6.

	Descriptive success %						*Predictive success %*					
	Neighbour Joining		**Best Match**		**BLOG**		**Neighbor Joining**		**Best match**		**BLOG**	
	avg	*std*	*avg*	*std*	*avg*	*std*	*avg*	*std.dev*	*avg*	*std*	*avg*	*std*
Ne1000	**89.34**	3.74	**91.80**	3.54	**92.04**	3.44	**89.94**	3.66	**92.52**	3.17	**92.17**	3.40
Ne10000	**81.56**	4.38	**88.44**	3.36	**92.30**	2.23	**88.60**	2.54	**92.27**	1.90	**93.26**	1.94
Ne50000	**42.84**	4.06	**73.26**	4.35	**91.06**	2.42	**83.10**	1.95	**91.27**	1.40	**90.83**	1.68

Table 7 *Success scores (%, N=100) of the compared methods when applied to DNA Barcode sequence datasets simulated according to different effective population sizes (Ne)*

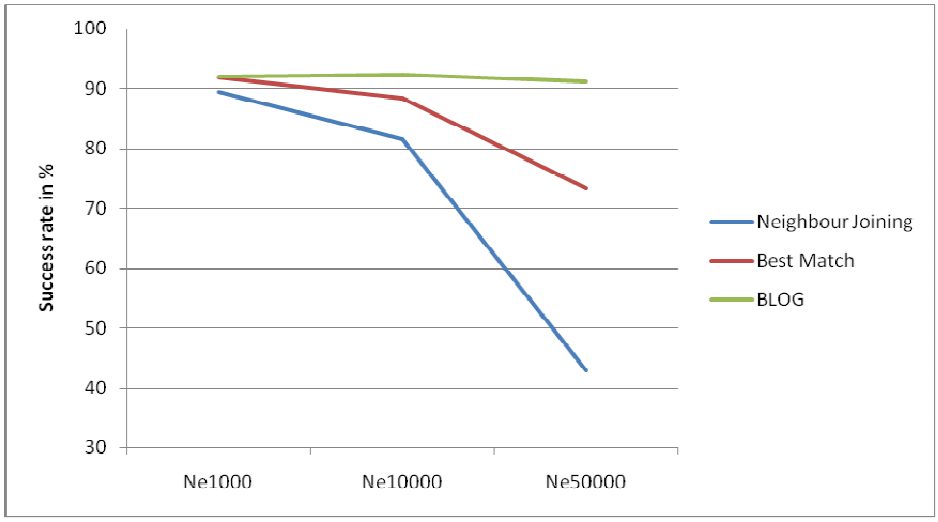

Figure 5 *Descriptive success rates*

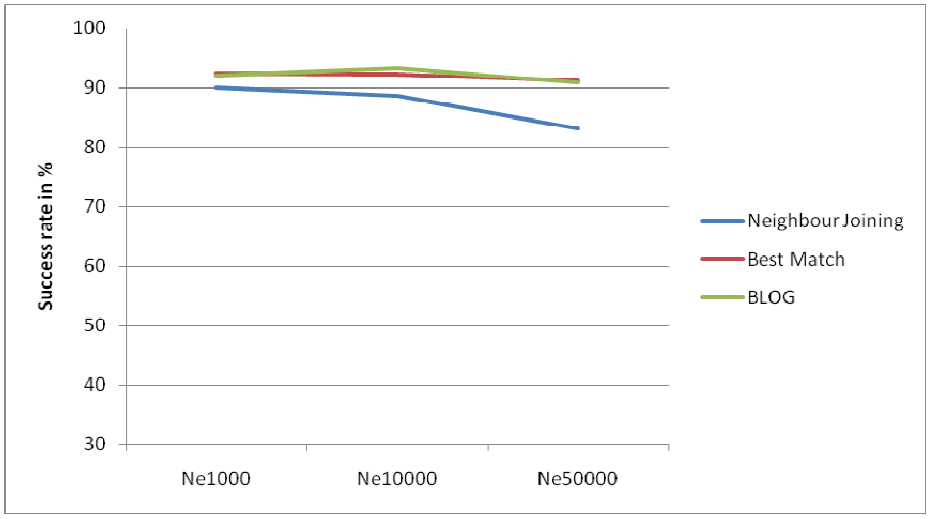

Figure 6 *Predictive success rates*

The comparative study shows that success scores generally decrease with increasing effective population size (Ne): Datasets that were simulated according to the lowest Ne (Ne=1000) had the highest average success score (91%). With an average success score of 78% datasets that were simulated according to the highest Ne (Ne=50000) were the most challenging in terms of species identification. We note that Best Match is the least stringent method of all: it simply matches a query sequence to the train sequence with highest similarity. In the absence of missing classes probably no other method can perform better. Of the three methods for DNA Barcode classification neighbor joining performed worst in terms of both descriptive and predictive power. BLOG performed best in terms of description of train datasets (89%), whereas Best Match and BLOG performed best in terms of predicting the right species membership of query sequences in the test datasets (92%). Taking both measures together, BLOG (90%) outperformed all methods.

4. Conclusions

In this work three different DNA Barcode data analysis methods, Neighbour Joining [2], Best Match [3] and BLOG [5], were presented and compared, using real and ad hoc simulated data sets. Neighbour Joining and Best Match are lazy learning distance based method. Neighbour Joining is a bottom-up clustering method used for the construction of phylogenetic trees based on sequence distance. Best Match is a lazy learning distance DNA barcode data analysis method. BLOG is a logic data mining approach for classifying species with DNA Barcode sequences. The aim of BLOG is to identify very few portions of the DNA sequence that are capable to identify the different species. With experimental result the approach has proven to be the most successful as it identifies logic formulas that effectively separate the different species with highest precision. Our comparison shows that BLOG performance is comparable to that of the Best Match method [3]. This method was shown to be the best performing one with complete reference datasets in various insect orders [4]. The distinctive advantage of BLOG is the output of the model, in terms of logic formulas, which gives a compact and precise description of the various species of the data set.

Authors' contributions

GF directed research. EW and GF designed BLOG methodology and software. RVV covered data simulation and suggested conceptual improvements of BLOG. RVV and EW performed the experiments. All authors contributed equally in writing and revising the paper.

Acknowledgements

We wish to thank Guido Drovandi for the precious contribution to the work through software engineering and scientific advices.

References

1. Hebert PDN, Cywinska A, Ball SL, deWaard JR: **Biological identifications through DNA barcodes**. *Proc R Soc B* 2003, **270**:313 - 321.

2. Saitou N, Nei M: **The Neighbour-joining method: a new method for reconstructing phylogenetic trees**. *Mol Biol Evol* 1987, **4**:406 - 425.

3. Rudolf Meier, Kwong Shiyang, Gaurav Vaidya, Peter K. L. NG: **DNA Barcoding and Taxonomy in Diptera: A Tale of High Intraspecific Variability and Low Identification Success**. *Systematic Biology* 2006, **55(5)**:715-728.

4. Massimiliano Virgilio, Thierry Backeljau, Bruno Nevado, Marc De Meyer: **Comparative performances of DNA barcoding across insect orders**. 2010, **206(11)**:4567-4573.

5. Paola Bertolazzi, Giovanni Felici, Emanuel Weitschek: **Learning to classify species with barcodes**. *BMC Bioinformatics* 2009, **10**:1-12.

6. **Mesquite Manual** [http://mesquiteproject.org/mesquite_folder/docs/mesquite/manual.html].

7. G.U. Yule: **A mathematical theory of evolution based on the conclusions of Dr. J. C. Willis**. *Philosophical Transactions of the Royal Society B: Biological Sciences* 1924, **213**:21–87.

8. E.F.Harding: **The probabilities of rooted tree-shapes generated by random bifurcation**. *Advances in Applied Probability* 1971, **3**:44–77.

9. David Posada: **jModelTest: Phylogenetic Model Averaging**. *Molecular Biology and Evolution* 2008, **7(25)**:1253-1256.

10. ORP Bininda-Emonds: *SeqCleaner.pl. Version 1.0.2.* AG Systematik und Evolutionsbiologie, IBU - Fakultät V, Carl von Ossietzky Universität Oldenburg,; 2010.

11. Paola Bertolazzi, Giovanni Felici, Paola Festa: **Logic based methods for SNPs tagging and reconstruction**. *Computers and Operations Research* 2010, **37**:1419-1426.

12. M. Charikar, V. Guruswami, R. Kumar, S. Rajagopalan, A. Sahai: **Combinatorial feature selection problems**. *41st Annual Symposium on Foundations of Computer Science* 2000:631.

13. M. Garey, D. Johnson: *Computers and Intractability*. W.H. Freeman; 1979.

14. Ivan Arisi, Mara D'Onofrio, Rossella Brandi, Armando Felsani, Simona Capsoni, Guido Drovandi, Giovanni Felici, Emanuel Weitschek, Paola Bertolazzi, Antonino Cattaneo: **Gene expression biomarkers in the brain of a mouse model for Alzheimer's Disease: mining of microarray data by logic classification and feature selection**. *Journal of Alzheimer's Disease* 2011, **24**.

15. Paola Bertolazzi, Giovanni Felici, Paola Festa, Giuseppe Lancia: **Logic classsification and feature selection for biomedical data**. *Computers and Mathematics with Applications* 2008, **55(5)**:889-899.

16. Thomas A. Feo, Mauricio G. C. Resende: **A probabilistic heuristic for a computationally difficult set covering problem**. *Operations Research Letters* 1989, **8**:67-71.

17. Giovanni Felici, Klaus Truemper: **A Minsat Approach for Learning in Logic Domains**. *INFORMS Journal on Computing* 2002, **14(1)**:20-36.

18. R Development Core Team: **R: A Language and Environment for Statistical Computing**. *R Foundation for Statistical Computing* 2008, **1**.

19. Emmanuel Paradis, Korbinian Strimmer: **APE: Analyses of Phylogenetics and Evolution in R language**. *Bioinformatics* 2003, **20**:289-290.