On the Homogenization of Data from Two Laboratories using Genetic Programming

Jose G. Moreno-Torres[1], Xavier Llorà[2], David E. Goldberg[3] and Rohit Bhargava[4]

[1] Department of Computer Science and Artificial Intelligence, Universidad de Granada, 18071 Granada, Spain
jose.garcia.mt@decsai.ugr.es
[2] National Center for Supercomputing Applications (NCSA) University of Illinois at Urbana-Champaign 1205 W. Clark Street, Urbana, Illinois, USA
xllora@illinois.edu
[3] Illinois Genetic Algorithms Laboratory (IlliGAL) University of Illinois at Urbana-Champaign 104 S. Mathews Ave, Urbana, Illinois, USA
deg@illinois.edu
[4] Department of Bioengineering University of Illinois at Urbana-Champaign 405 N. Mathews Ave, Urbana, Illinois, USA
rbx@uiuc.edu

Abstract. In experimental sciences, diversity tends to difficult predictive models' proper generalization across data provided by different laboratories. Thus, training on a data set produced by one lab and testing on data provided by another lab usually results in low classification accuracy. Despite the fact that the same protocols were followed, variability on measurements can introduce unforeseen variations that affect the quality of the model. This paper proposes a Genetic Programming based approach, where a transformation of the data from the second lab is evolved driven by classifier performance. A real-world problem, prostate cancer diagnosis, is presented as an example where the proposed approach was capable of repairing the fracture between the data of two different laboratories.

1 Introduction

The assumption that a properly trained classifier will be able to predict the behavior of unseen data from the same problem is at the core of any automatic classification process. However, this hypothesis tends to prove unreliable when dealing with biological data (or other experimental sciences), especially when such data is provided by more than one laboratory, even if they are following the same protocols to obtain it.

This paper presents an example of such a case, a prostate cancer diagnosis problem where a classifier built using the data of the first laboratory performs

very accurately on the test data from that same laboratory, but comparatively poorly on the data from the second one. It is assumed that this behavior is due to a fracture between the data of the two laboratories, and a Genetic Programming (GP) method is developed to homogenize the data in subsequent subsets. We consider this method a form of feature extraction because the new dataset is constructed with new features which are functional mappings of the old ones.

The method presented in this paper attempts to optimize a transformation over the data from the second laboratory, in terms of classifier performance. That is, the data from the second lab is transformed into a new dataset where the classifier, trained on the data from the first lab, performs as accurately as possible. If the performance achieved by the classifier in this new, transformed, dataset, is equivalent to the one obtained in the data from the first lab, we understand the data has been homogenized.

More formally, the classifier f is trained on data from one laboratory (dataset A), such that $y = f(xA)$ is the class prediction for one instance xA of dataset A. For the data from the other lab (dataset B), it is assumed that there exists a transformation T such that $f(T(xB))$ is a good classifier for instances xB of dataset B. The 'goodness' of the classifier is measured by the loss function $l(f(T(xB)), y)$, where y is the class associated with xB, and $l(.,.)$ is a measure of distance between $f(T(xB))$ and y. The aim is to find a transformation T such that the average loss over all instances in B is minimized.

The remainder of this paper is organized as follows: In Section 2, some preliminaries about the techniques used and some approaches to similar problems in the literature are presented. Section 3 has a description of the proposed algorithm. Section 4 details the real-world biological dataset that motivates this paper. Section 5 includes the experimental setup, along with the results obtained, and an analysis. Finally, some concluding remarks are made in Section 6.

2 Preliminaries

This section is divided in the following way: In Section 2.1 we introduce the notation that has been used in this paper. Then we include a brief summary of what has been done in feature extraction in Section 2.2, and a short review of the different approaches we found in the specialized literature on the use of GP for feature extraction in Section 2.3.

2.1 Notation

When describing the problem, datasets A, B and S correspond to:

- A: The original dataset, provided by the first lab, that was used to build the classifier.
- B: The problem dataset, from the second lab. The classifier is not accurate on this dataset, and that is what the proposed algorithm attempts to solve.
- S: The solution dataset, result of applying the evolved transformation to the samples in dataset B. The goal is to have the classifier performance be as high as possible on this dataset.

2.2 Feature Extraction

Feature extraction is one form of pre-processing, which creates new features as functional mappings of the old ones. An early proposer of such a term was probably Wyse in 1980 [1], in a paper about intrinsic dimensionality estimation. There are multiple techniques that have been applied to feature extraction throughout the years, ranging from principal component analysis (PCA) to support vector machines (SVMs) to GAs (see [2–4], respectively, for some examples).

Among the foundations papers in the literature, Liu's book in 1998 [5] is one of the earlier compilations of the field. A workshop held in 2003 [6], led Guyon & Elisseeff to publish a book with an important treatment of the foundations of feature extraction[7].

2.3 Genetic Programming-based Feature Extraction

Genetic Programming (GP) has been used extensively to optimize feature extraction and selection tasks. One of the first contributions in this line was the work published by Tackett in 1993 [8], who applied GP to feature discovery and image discrimination tasks.

We can consider two main branches in the philosophy of GP-based feature extraction:

1 On one hand, we have the proposals that focus only on the feature extraction procedure, of which there are multiple examples: Sherrah et al. [9] presented in 1997 the evolutionary pre-processor (EPrep), which searches for an optimal feature extractor by minimizing the misclassification error over three randomly selected classifiers. Kotani et al.'s work from 1999 [10] determined the optimal polynomial combinations of raw features to pass to a k-nearest neighbor classifier. In 2001, Bot [11] evolved transformed features, one-at-a-time, again for a k-NN classifier, utilizing each new feature only if it improved the overall classification performance. Zhang & Rockett, in 2006, [12] used multiobjective GP to learn optimal feature extraction in order to fold the high-dimensional pattern vector to a one-dimensional decision space where the classification would be trivial. Lastly, also in 2006, Guo & Nandi [13] optimized a modified Fisher discriminant using GP, and then Zhang & Rockett [14] extended their work by using a multiobjective approach to prevent tree bloat.
2 On the other hand, some authors have chosen to evolve a full classifier with an embedded feature extraction step. As an example, Harris [15] proposed in 1997 a co-evolutionary strategy involving the simultaneous evolution of the feature extraction procedure along with a classifier. More recently, Smith & Bull [16] developed a hybrid feature construction and selection method using GP together with a GA.

2.4 Finding and repairing fractures between data

Among the proposals to quantify the fracture in the data, we would like to mention the one by Wang et al. [17], where the authors present the idea of

correspondence tracing. They propose an algorithm for the discovering of changes of classification characteristics, which is based on the comparison between two rule-based classifiers, one built from each dataset. Yang et al. [18] presented in 2008 the idea of conceptual equivalence as a method for contrast mining, which consists of the discovery of discrepancies between datasets. Lately, it is important to mention the work by Cieslak and Chawla [19], which presents a statistical framework to analyze changes in data distribution resulting in fractures between the data.

The fundamental difference between the mentioned works and this one is we focus on repairing the fracture by modifying the data, using a general method that works with any kind of data fracture, while they propose methods to quantify said fracture that work provided some conditions.

3 A proposal for GP-based feature extraction to Homogenize Data from Two Laboratories

The problem we are attempting to solve is the design of a method that can create a transformation from a dataset (dataset B) where a classification model built using the data from a different dataset (dataset A) is not accurate; into a new dataset (dataset S) where the classifier is more accurate. Said classifier is kept unchanged throughout the process.

We decided to use GP to solve the problem for a number of reasons:

1 It is well suited to evolve arbitrary expressions because its chromosomes are trees. This is useful in our case because we want to have the maximum possible flexibility in terms of the functional expressions of this transformations.
2 GP provides highly-interpretable solutions. This is an advantage because our goal is not only to have a new dataset where the classifier works, but also to analyze what was the problem in the first dataset.

Once GP was chosen, we needed to decide what terminals and operators to use, how to calculate the fitness of an individual and which evolutionary parameters (population size, number of generations, selection and mutation rates, etc) are appropriate for the problem at hand.

3.1 Solutions representation: Context-free grammar

The representation of the solutions was achieved by extending GP to evolve more than one tree per solution. Each individual is composed by n trees, where n is the number of attributes present in the dataset. We are trying to develop a new dataset with the same number of attributes as the old one, since this new dataset needs to be fed to the existing model. In the tree structure, the leaves are either constants (we use the Ephemeral Random Constant approach [20]) or attributes from the original dataset. The intermediate nodes are functions from the function set, which is specific to each problem.

The attributes on the transformed dataset are represented by algebraic expressions. These expressions are generated according to the rules of a context-free grammar which allows the absence of some of the functions or terminals. The grammar corresponding to the example problem would look like this:

$$
\begin{aligned}
&Start \rightarrow Tree\ Tree \\
&Tree \rightarrow Node \\
&Node \rightarrow Node\ Operator\ Node \\
&Node \rightarrow Terminal \\
&Operator \rightarrow + \mid - \mid * \mid \div \\
&Terminal \rightarrow x_0 \mid x_1 \mid E \\
&E \rightarrow realNumber(represented\ by\ e)
\end{aligned}
$$

3.2 Fitness evaluation

The fitness evaluation procedure is probably the most treated aspect of design in the literature when dealing with GP-based feature extraction. As has been stated before, the idea is to have the provided classifier's performance drive the evolution. To achieve that, our method calculates fitness as the classifier's accuracy over the dataset obtained by applying the transformations encoded in the individual (training-set accuracy).

3.3 Genetic operators

This section details the choices made for selection, crossover and mutation operators. Since the objective of this work is not to squeeze the maximum possible performance from GP, but rather to show that it is an appropriate technique for the problem and that it can indeed solve it, we did not pay special attention to these choices, and picked the most common ones in the specialized literature.

- Tournament selection without replacement. To perform this selection, s individuals are first randomly picked from the population (where s is the tournament size), while avoiding using any member of the population more than once. The selected individual is then chosen as the one with the best fitness among those picked in the first stage.
- One-point crossover: A subtree from one of the parents is substituted by one from the other parent. This procedure is carried over in the following way:
  1. Randomly select a non-root non-leave node on each of the two parents.
  2. The first child is the result of swapping the subtree below the selected node in the father for that of the mother.
  3. The second child is the result of swapping the subtree below the selected node in the mother for that of the father.
- Swap mutation: This is a conservative mutation operator, that helps diversify the search within a close neighborhood of a given solution. It consists of exchanging the primitive associated to a node by one that has the same number of arguments.

– Replacement mutation: This is a more aggressive mutation operator that leads to diversification in a larger neighborhood. The procedure to perform this mutation is the following:
  1 Randomly select a non-root non-leave node on the tree to mutate.
  2 Create a random tree of depth no more than a fixed maximum depth. In this work, the maximum depth allowed was 5.
  3 Swap the subtree below the selected node for the randomly generated one.

3.4 Function set

Which functions to include in the function set are usually dependent on the problem. Since one of our goals is to have an algorithm as universal and robust as possible, where the user does not need to fine-tune any parameters to achieve good performance; we decided not to study the effect of different function set choices. We chose the default functions most authors use in the literature: $\{+, -, *, \div, exp, cos\}$.

3.5 Parameters

Table 1 summarizes the parameters used for the experiments.

Table 1. Evolutionary parameters for a n_v-dimensional problem

Parameter	Value
Number of trees	n_v
Population size	$400 * n_v$
Duration of the run	100 generations
Selection operator	Tournament without replacement
Tournament size	$log_2(n_v) + 1$
Crossover operator	One-point crossover
Crossover probability	0.9
Mutation operator	Replacement & Swap mutations
Replacement mutation probability	0.001
Swap mutation probability	0.01
Maximum depth of the swapped in subtree	5
Function set	$\{+, -, *, \div, cos, exp\}$
Terminal set	$\{x_0$,x_1,...,$x_{n_v} - 1$, e$\}$

3.6 Execution flow

Algorithm 1 contains a summary of the execution flow of the GP procedure, which follows a classical evolutionary scheme. It stops after a user-defined number of generations,

Algorithm 1 Execution flow of the GP method

```
1. Randomly create the initial population by applying the
   context-free grammar in Section 3.1.
2. Repeat Ng times (where Ng is the number of generations)
  2.1 Evaluate the current population, using the procedure
      seen in Section 3.2.
  2.2 Apply selection and crossover to create a new
      population that will replace the old one.
  2.3 Apply the mutation operators to the new population.
3. Return the best individual ever seen.
```

4 Case Study: Prostate Cancer Diagnosis

Prostate cancer is the most common non-skin malignancy in the western world. The American Cancer Society estimated 192,280 new cases and 27,360 deaths related to prostate cancer in 2009 [21]. Recognizing the public health implications of this disease, men are actively screened through digital rectal examinations and/or serum prostate specific antigen (PSA) level testing. If these screening tests are suspicious, prostate tissue is extracted, or biopsied, from the patient and examined for structural alterations. Due to imperfect screening technologies and repeated examinations, it is estimated that more than one million people undergo biopsies in the US alone.

4.1 Diagnostic procedure

Biopsy, followed by manual examination under a microscope is the primary means to definitively diagnose prostate cancer as well as most internal cancers in the human body. Pathologists are trained to recognize patterns of disease in the architecture of tissue, local structural morphology and alterations in cell size and shape. Specific patterns of specific cell types distinguish cancerous and non-cancerous tissues. Hence, the primary task of the pathologist examining tissue for cancer is to locate foci of the cell of interest and examine them for alterations indicative of disease. A detailed explanation of the procedure is beyond the scope of this paper and can be found elsewhere [22–25].

Operator fatigue is well-documented and guidelines limit the workload and rate of examination of samples by a single operator (examination speed and throughput). Importantly, inter- and intra-pathologist variation complicates decision making. For this reason, it would be extremely interesting to have an accurate automatic classifier to help reduce the load on the pathologists. This was partially achieved in [24], but some issues remain open.

4.2 The generalization problem

Llorà et al. [24] successfully applied a genetics-based approach to the development of a classifier that obtained human-competitive results based on FTIR

data. However, the classifier built from the data obtained from one laboratory proved remarkably inaccurate when applied to classify data from a different hospital. Since all the experimental procedure was identical; using the same machine, measuring and post-processing; and having the exact same lab protocols, both for tissue extraction and staining; there was no factor that could explain this discrepancy.

What we attempt to do with this work is develop an algorithm that can evolve a transformation over the data from the second laboratory, creating a new dataset where the classifier built from the first lab is as accurate as possible.

4.3 Pre-processing of the data

The biological data obtained from the laboratories has an enormous size (in the range of 14GB of storage per sample); and parallel computing was needed to achieve better-than-human results. For this reason, feature selection was performed on the dataset obtained by FTIR. It was done by applying an evaluation of pairwise error and incremental increase in classification accuracy for every class, resulting in a subset of 93 attributes. This reduced dataset provided enough information for classifier performance to be rather satisfactory: a simple C4.5 classifier achieved $\sim 95\%$ accuracy on the data from the first lab, but only $\sim 80\%$ on the second one. The dataset consists of 789 samples from one laboratory and 665 from the other one. These samples represent 0.01% of the total data available for each data set, which were selected applying stratified sampling without replacement. A detailed description of the data pre-processing procedure can be found in [22].

The experiments reported in this paper were performed utilizing the reduced dataset, since the associated computational costs make it unfeasible to work with the complete one. The reduced dataset is made of 93 real attributes, and there are two classes (positive and negative diagnosis). The dataset consists of 789 samples from one laboratory and 665 from the other one, with a $60\% - 40\%$ class distribution.

5 Experimental Study

This section is organized in the following way: To begin with, a general description of the experimental procedure is presented in Section 5.1, and the parameters used for the experiment. The results obtained are presented in Section 5.2, a statistical analysis is shown in Section 5.3, and lastly some sample transformations are shown in Section 5.4.

5.1 Experimental Framework

The experimental methodology can be summarized as follows:

1 Consider each of the provided datasets (one from each lab) to be datasets A and B respectively.

2 From dataset A, build a classifier. We chose C4.5 [26], but any other classifier would work exactly the same; due to the fact that the proposed method uses the learned classifier as a black box.
3 Apply our method to dataset B in order to evolve a transformation that will create a solution dataset S. Use 5-fold cross validation over dataset S, so that training and test set accuracy results can be obtained.
4 Check the performance of the step 2 classifier on dataset S. Ideally, it should be close to the one on dataset A, meaning the proposed method has successfully discovered the hidden transformation and inverted it.

5.2 Performance results

This section presents the results for the Prostate Cancer problem, in terms of classifier accuracy. The results obtained can be seen in table 2.

Table 2. Classifier performance results

Classifier performance in dataset ...				
A-training	A-test	B	S-training	S-test
0.95435	0.92015	0.83570	0.95191	0.92866

The performance results are promising. First and foremost, the proposed method was able to find a transformation over the data from the second laboratory that made the classifier work just as well as it did on the data from the first lab, effectively finding the fracture in the data (that is, the difference in data distribution between the data sets provided by the two labs) that prevented the classifier from working accurately.

5.3 Statistical analysis

To complete the experimental study, we performed a statistical comparison between the classifier performance over datasets A, B and S.

In [27–30] a set of simple, safe and robust non-parametric tests for statistical comparisons of classifiers are recommended. One of them is the Wilcoxon Signed-Ranks Test [31, 32], which is the test that we have selected to do the comparison.

In order to perform the Wilcoxon test, we used the results from each partition in the 5-fold cross validation procedure. We ran the experiment four times, resulting in $4 * 5 = 20$ performance samples to carry out the statistical test. R^+ corresponds to the first algorithm in the comparison winning, R^- to the second one.

Table 3. Wilcoxon signed-ranks test results

Comparison	R^+	R^-	p-value	null hypothesis of equality
A-test vs B	210	0	$1.91E-007$	*rejected* (A-test outperforms B)
B vs S-test	0	210	$1.91E-007$	*rejected* (S-test outperforms B)
A-training vs S-training	126	84	$--$	*accepted*
A-test vs S-test	84	126	$--$	*accepted*

We can conclude our method has proved to be capable of fully homogenizing the data from both laboratories regarding classifier performance, both in terms of training and test set.

5.4 Obtained transformations

Figure 1 contains a sample of some of the evolved expressions for the best individual found by our method. Since the dataset has 93 attributes, the individual was composed of 93 trees, but for space concerns only the attributes relevant to the C4.5 classifier were included here.

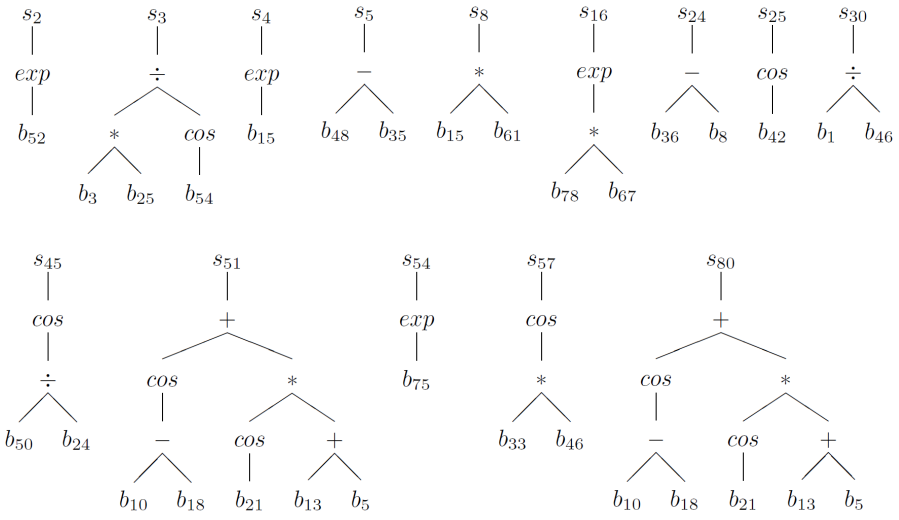

Fig. 1. Tree representation of the expressions contained in a solution to the Prostate Cancer problem

6 Concluding remarks

We have presented a new algorithm that approaches a common problem in real life for which not many solutions have been proposed in evolutionary computing. The problem in question is the repairing of fractures between data by adjusting the data itself, not the classifiers built from it.

We have developed a solution to the problem by means of a GP-based algorithm that performs feature extraction on the problem dataset driven by the accuracy of the previously built classifier.

We have applied our method to a real-world problem where data from two different laboratories regarding prostate cancer diagnosis was provided, and where the classifier learned from one did not perform well enough on the other. Our algorithm was capable of learning a transformation over the second dataset that made the classifier fit just as well as it did on the first one. The validation results with 5-fold cross validation also support the idea that the algorithm is obtaining good results; and has a strong generalization power.

We have applied a statistical analysis methodology that supports the claim that the classifier performance obtained on the solution dataset significantly outperforms the one obtained on the problem dataset.

Lastly, we have shown the learned transformations. Unfortunately, we have not been able to extract any useful information from them yet.

Acknowledgments

Jose García Moreno-Torres was supported by a scholarship from 'Obra Social la Caixa' and is currently supported by a FPU grant from the Ministerio de Educación y Ciencia of the Spanish Government and the KEEL project. Rohit Bhargava would like to acknowledge collaborators over the years, especially Dr. Stephen M. Hewitt and Dr. Ira W. Levin of the National Institutes of Health, for numerous useful discussions and guidance. Funding for this work was provided in part by University of Illinois Research Board and by the Department of Defense Prostate Cancer Research Program. This work was also funded in part by the National Center for Supercomputing Applications and the University of Illinois, under the auspices of the NCSA/UIUC faculty fellows program.

References

1. Wyse, N., Dubes, R., Jain, A.: A critical evaluation of intrinsic dimensionality algorithmsa critical evaluation of intrinsic dimensionality algorithms. In Gelsema, E.S., Kanal, L.N., eds.: Pattern recognition in practice, Amsterdam, Morgan Kauffman Publishers, Inc. (1980) 415–425
2. Kim, K.A., Oh, S.Y., Choi, H.C.: Facial feature extraction using pca and wavelet multi-resolution images. In: Sixth IEEE International Conference on Automatic Face and Gesture Recognition, Los Alamitos, CA, USA, IEEE Computer Society (2004) 439

3. Podolak, I.T.: Facial component extraction and face recognition with support vector machines. In: FGR '02: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, Washington, DC, USA, IEEE Computer Society (2002) 83
4. Pei, M., Goodman, E.D., Punch, W.F.: Pattern discovery from data using genetic algorithms. In: Proceeding of 1st Pacific-Asia Conference Knowledge Discovery & Data Mining(PAKDD-97). (1997)
5. Liu, H., Motoda, H.: Feature extraction, construction and selection : a data mining perspective. Volume SECS 453. Kluwer Academic, Boston (1998)
6. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3** (2003) 1157–1182
7. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L., eds.: Feature Extraction, Foundations and Applications. Springer (2006)
8. Tackett, W.A.: Genetic programming for feature discovery and image discrimination. In: Proceedings of the 5th International Conference on Genetic Algorithms, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1993) 303–311
9. Sherrah, J.R., Bogner, R.E., Bouzerdoum, A.: The evolutionary pre-processor: Automatic feature extraction for supervised classification using genetic programming. In: Proc. 2nd International Conference on Genetic Programming (GP-97, Morgan Kaufmann (1997) 304–312
10. Kotani, M., Ozawa, S., Nakai, M., Akazawa, K.: Emergence of feature extraction function using genetic programming. In: KES. (1999) 149–152
11. Bot, M.C.J.: Feature extraction for the k-nearest neighbour classifier with genetic programming. In: EuroGP '01: Proceedings of the 4th European Conference on Genetic Programming, London, UK, Springer-Verlag (2001) 256–267
12. Zhang, Y., Rockett, P.I.: A generic optimal feature extraction method using multiobjective genetic programming. Technical Report VIE 2006/001, Department of Electronic and Electrical Engineering, University of Sheffield, UK (2006)
13. Guo, H., Nandi, A.K.: Breast cancer diagnosis using genetic programming generated feature. Pattern Recognition **39**(5) (2006) 980–987
14. Zhang, Y., Rockett, P.I.: A generic multi-dimensional feature extraction method using multiobjective genetic programming. Evolutionary Computation **17**(1) (2009) 89–115
15. Harris, C.: An investigation into the Application of Genetic Programming techniques to Signal Analysis and Feature Detection. PhD thesis, University College, London (26 September 1997)
16. Smith, M.G., Bull, L.: Genetic programming with a genetic algorithm for feature construction and selection. Genetic Programming and Evolvable Machines **6**(3) (2005) 265–281
17. Wang, K., Zhou, S., Fu, C.A., Yu, J.X., Jeffrey, F., Yu, X.: Mining changes of classification by correspondence tracing. In: In Proceedings of the 2003 SIAM International Conference on Data Mining (SDM 2003). (2003)
18. Yang, Y., Wu, X., Zhu, X.: Conceptual equivalence for contrast mining in classification learning. Data & Knowledge Engineering **67**(3) (2008) 413–429
19. Cieslak, D.A., Chawla, N.V.: A framework for monitoring classifiers' performance: when and why failure occurs? Knowledge and Information Systems **18**(1) (2009) 83–108
20. Koza, J.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. The MIT Press, Cambridge, MA (1992)

21. AmericanCancerSociety: How many men get prostate cancer? http://www.cancer.org/docroot/CRI/content/CRI_2_2_1X_How_many_men_get_prostate_cancer_36.asp
22. Fernandez, D.C., Bhargava, R., Hewitt, S.M., Levin, I.W.: Infrared spectroscopic imaging for histopathologic recognition. Nature biotechnology **23**(4) (2005) 469–474
23. Levin, I.W., Bhargava, R.: Fourier transform infrared vibrational spectroscopic imaging: integrating microscopy and molecular recognition. Annual Review of Physical Chemistry **56** (2005) 429–74
24. Llorà, X., Reddy, R., Matesic, B., Bhargava, R.: Towards better than human capability in diagnosing prostate cancer using infrared spectroscopic imaging. In: GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation, New York, NY, USA, ACM (2007) 2098–2105
25. Llorà, X., Priya, A., Bhargava, R.: Observer-invariant histopathology using genetics-based machine learning. Natural Computing: an international journal **8**(1) (2009) 101–120
26. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
27. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7** (2006) 1–30
28. García, S., Herrera, F.: An extension on 'statistical comparisons of classifiers over multiple data sets' for all pairwise comparisons. Journal of Machine Learning Research **9** (2008) 2677–2694
29. García, S., Fernández, A., Luengo, J., Herrera, F.: A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability. Soft Computing **13**(10) (2009) 959–977
30. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Information Sciences **180**(10) (2010) 2044–2064
31. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bulletin **1**(6) (1945) 80–83
32. Sheskin, D.J.: Handbook of Parametric and Nonparametric Statistical Procedures (4th Edition). Chapman & Hall/CRC (2007)