Assessing the Impact of Changing environments on Classifier Performance *

Rocío Alaiz-Rodríguez[1] and Nathalie Japkowicz[2]

[1] Dpto. de Ingeniería Eléctrica y de Sistemas y Automática,
Campus de Vegazana, Universidad de León, 24071 León, Spain,
rocio.alaiz@unileon.es
[2] SITE. University of Ottawa.
150 Louis Pasteur, P.O. Box 450 Stn. A Ottawa, Ontario, Canada,
nat@site.uottawa.ca

Abstract. The purpose of this paper is to test the hypothesis that simple classifiers are more robust to changing environments than complex ones. We propose a strategy for generating artificial, but realistic domains, which allows us to control the changing environment and test a variety of situations. Our results suggest that evaluating classifiers on such tasks is not straightforward since the changed environment can yield a simpler or more complex domain. We propose a metric capable of taking this issue into consideration and evaluate our classifiers using it. We conclude that in mild cases of population drifts simple classifiers deteriorate more than complex ones and that in more severe cases as well as in class definition changes, all classifiers deteriorate to about the same extent. This means that in all cases, complex classifiers remain more accurate than simpler ones, thus challenging the hypothesis that simple classifiers are more robust to changing environments than complex ones.

1 Introduction.

A common assumption in supervised learning is that training and future data come from the same, although unknown, distribution. This fundamental assumption, however, does not often hold in practice [3] and this may lead to a significant performance deterioration in the induced classifiers.

Several approaches have been proposed so far in order to deal with different mismatches between training data and data drawn from the real operating conditions. They either follow an adaptive strategy, relying on an unlabeled dataset representative of the new conditions to update the classifier accordingly [10] or they deal with this uncertainty problem by using a particularly robust approach [1]. An important claim made by David Hand in [3] says that, because in real world domains the distributions involved in the training set are often not representative of future data, performance gains reached by more sophisticated

* Supported by the Natural Science and Engineering Council of Canada and the Spanish MEC project DPI2006-02550.

classifiers that are able to model small idiosyncrasies found in the underlying distribution of the training set are marginal with respect to the performance of simple standard classifiers.This paper focuses on this hypothesis and analyzes whether complex classifiers keep their advantage at testing time once the distributions have shifted, or as suggested in [3], simpler classifiers that focus on the more general features of the training set distribution—which will persist throughout the distributional shift occurring between training and testing time—will be more robust than (or at least as robust as) the more sophisticated classifiers. In this work, the complexity concept refers to the classifier capability to define arbitrarily complex decision frontiers.

To our knowledge, there has previously been no such comparative studies of the robustness of different classifiers against these mismatches between training and test data. Although Hand's work [3] illustrates the performance of two classifiers (LDA and a tree model) in a bank application over time, no detailed description of the experimental methods is provided and there is no chance to refer to the ideal scenario. We set out to investigate this hypothesis using a simulated medical domain whose distributions we were able to control fully. In particular, we generated a number of variations of our standard domain, relying on common-sense scenarios and distribution changes studied in the literature, and tested four different classifiers on all these domains. The classifiers range from simple models such as 1R rules or simple Neural Networks to more sophisticated models such as Decision Trees and Neural Networks with a more complex architecture.

Experimental results show different trends in the relative performance between simple and more sophisticated classifiers. Although we observed that under some of the assessed changing scenarios, as suggested by Hand [3], the superiority of the complex classifiers over the simple ones is not that significant, we also found that this does not necessarily imply that simple classifiers are less susceptible to performance deterioration. In fact, their apparent robustness may be due to a simplification in the classification task.

The remainder of this article is organized as follows. Sect. 2 summarizes the distribution changes studied in the literature. Sect. 3 describes our hypothesis testing methodology, detailing the domain generator we designed and the variations we applied to it. Sect. 4 shows our experimental results and discusses their implications. Sect. 5 concludes with a summary and suggestions for future work.

2 Changing Environments.

Consider a standard supervised classification problem with a labeled data set $S = \{(\mathbf{x}^k, \mathbf{d}^k), k = 1, \ldots, K\}$ and examples independently drawn from an unknown distribution, where $\mathbf{x}^k$ is an observation feature vector and $\mathbf{d}^k$ is the class label.

The fundamental assumption of supervised learning is that the joint probability distribution $p(\mathbf{x}, \mathbf{d})$ will remain unchanged between training and testing. There are, however, some mismatches that are likely to appear in practice and have already been studied in the literature.

One of the most studied mismatches is a situation where the conditional probability $p(\mathbf{d}|\mathbf{x})$ remains unchanged, but the input distribution $p(\mathbf{x})$ differs from training to future data. This has been referred to as *population drift* [7, 3], *covariate shift* [11, 13] or *sample selection bias* [5, 3], although this last term in fact refers to the design data acquisition. Such mismatches can be encountered in practical fields such as banking applications, medical diagnosis or bioinformatics. In this work we will refer to it as *population drift*.

Another kind of distribution change is the *class definition change* where $p(\mathbf{x})$ is not altered but, $p(\mathbf{d}|\mathbf{x})$ varies from training to test [3]. The term *concept drift* is also used to refer to this variation [8] as well as *functional relation change* [13]. In this work, we will use the term *class definition change* to refer to this change.

Another widely studied scenario is the change in class distribution, where the class prior probability $p(\mathbf{d})$ varies from training to test, but $p(\mathbf{x}|\mathbf{d})$ remains unaltered. The problem of changing class distributions has been studied from different perspectives in [10, 1, 9, 2].

In the remainder of the paper will focus on the problems of population drift and class definition change and we will specifically focus on the following two issues: (a) Whether, in case of both, population drifts and class definition changes, an actual drop in performance can generally be observed by all kinds of classifiers, (b)Whether it is correct to assume that the simpler classifiers will maintain their performance more reliably than the more sophisticated ones in such cases.

3 Our Experimental framework.

In order to test the hypotheses put forth in [3], we generated an artificial domain, that we subsequently tested under various regimen of population drifts and class definition changes. We chose a simple domain that anyone could relate to in order to be able to consider rational rather than completely random variations. In order to make the problem interesting, we also introduced some instance of attribute dependency as well as a great deal of uncertainty, in various cases. All the domains were generated completely automatically following the attribute and class rules described in this section.

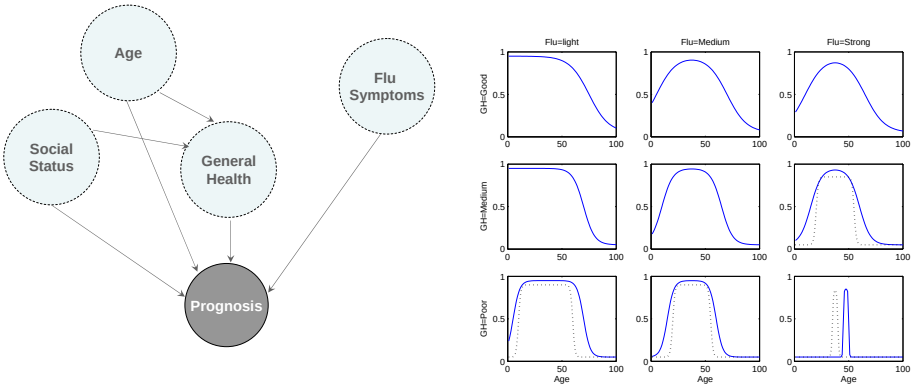
3.1 General setting.

Our domain is a simulated medical domain that states the prognosis of patients infected with the flu and described by the following attribute vector $\mathbf{x}$ where:

- x_1 is the patient's age (described by a discrete value [Infant, Teenager, YoungAdult, Adult, OldAdult, Elderly]).
- x_2 defines the severity of the flu symptoms (described by a discrete value [Light, Medium, Strong]).
- x_3 represents the patient's general health condition (described by a discrete value [Good, Medium, Poor]).
- x_4 represents the patient's social position (described by a discrete value [Rich, MiddleClass, Poor]).

The patient's classification is the patient's prognosis after a month (described by a nominal value: NormalRemission, Complications).

In order to make the problem interesting for classifiers, we assume that the features are not independent of one another and that certain feature values may be irrelevant. We assume that the dependency relations of the features are as shown in Fig.1(a), where each node represents a feature (or the class) and an arc between two nodes represents the dependency relation.



(a) (b)

Fig. 1. 1(a) Attribute dependency in our simulated domain. 1(b) Probability of Normal Remission as a function of Age, Flu Symptoms and General Health. When two lines appear, the discontinuous one applies to instances with Social Status equal to Poor.

3.2 Original Setting

We will consider several settings for our domain. In this section, we describe the original one that corresponds to a Negative Growth Population (NGP) with the age distribution shown in Table 1. The next subsections will focus on others.

The original setting reflects the following common-sense rules:

- Infants, old adults and the elderly are more prone to complications than people in other categories.
- Severe flu symptoms cause more complications than mild ones.
- People in poor general health are more prone to complications than generally healthier people.
- People at the bottom of the social ladder are likely to be in poorer general health than wealthier people, and are, thus, also more prone to complications.

We do not have the space necessary to describe the details of our data generator, but Figure 1 (b) illustrates the distribution of the NormalRemission class we obtain as a function of our four attributes.

Having described the original domain, we now turn to the various kinds of modifications that we considered. Three general situations will be studied: *Population drift*, *Population drift-NR with not Represented cases* and *Class definition change*. We now detail each of these situations.

Age Group	Prior Probability
Infant	0.07
Teenager	0.10
Young	0.36
Adult	0.24
Old Adult	0.17
Elderly	0.06

Table 1. Age group distribution for a Negative Growth Population (NGP).

Age Group	Prior Probability
Infant	0.30
Teenager	0.25
Young	0.20
Adult	0.12
Old Adult	0.08
Elderly	0.05

Table 2. Age group distribution for a Developing Population (DP).

Age Group	Prior Probability
Infant	0.14
Teenager	0.20
Young	0.20
Adult	0.20
Old Adult	0.14
Elderly	0.12

Table 3. Age group distribution for a Zero Growth Population (ZGP).

3.3 Population drift with full representation

The purpose of our experiments in population drifts with full representation is to test whether, indeed, changes in case proportions do not have any effect on classification accuracy. We investigated several scenarios, each plausible.

Developing population (DP). The population where the classifier is deployed corresponds to a developing country region with high birth rates and also high death rates. The age distribution for the Developing Population (DP) is given in Table 2.

Zero Growth Population (ZGP) The population has zero growth with age distribution given in Table 3.

Season changes (NGP/W). Flu symptoms get stronger. In the season change category, we investigated the scenario in which the original data set corresponds to a normal time, and the new data set corresponds to a rougher season (colder winter) during which the flu symptoms get stronger.

Season changes (NGP/SW). Flu symptoms get softer. This situation corresponds to a milder winter.

Season changes (NGP/DW). Drastic Winter-Flu symptoms get stronger and General health declines. In addition to the previous situation with strong winter, we are adding changes in the general health. In general, more cases of complications occur, with a greater markedness in poor people with poor health and elderly and infants.

Population is much poorer (NGP/P). Instead of the Normal distribution that we previously used for social status, we are considering a distribution skewed towards the poor class.

Population is much poorer and the winter is drastic (NGP/P+DW). The changes of situations in DW and P are implemented simultaneously.

3.4 Population drift-NR (Not Represented cases)

We consider several situations where an age group is missing in the training population set (No Infant, No Teenager, No Young, etc) or two groups (neither Infant nor Elderly, neither Teenager nor Old Adult, neither Infant nor Teenager, neither Old Adult nor Elderly). The remaining groups are represented proportionally to their original prior probability.

3.5 Class definition change

In this case, the population is exactly the same, but the labeling rules change. We assume that the test set was generated later, and there has been either fewer complications or more complications at that time.

Class definition change (MC). More complications. Class definition changes are defined so that the probability of Normal Remission decreases for certain ages, social statuses and flu symptoms.

Class definition change (FC). Fewer complications. Unlike the changes described above, parameters in the labeling rules have been modified in order to widen the age interval for which the probability of Normal remission is high.

4 Experimental Results.

The effect of changing conditions between training and test sets was assessed in both simple and sophisticated classifiers. As simple classifiers we used a *1R* classifier [4] and a *Simple NN* (Neural Network with a multilayer perceptron architecture and only 1 node in the hidden layer). As more complex classifiers, a C4.5 *Decision Tree* and *Complex NN* with a higher number of nodes (10, for this experiment) in the hidden layer so that, decision frontiers can be more complex.

Training and test sets with 1000 instances each one were generated and the results are the average over 30 different trials (with different data sets). Classifier training was carried out using Weka [12] with the following parameters selected for the NGP population: 1R with a minimum bucket size of 6 for discretizing numeric attributes, C4.5 decision tree with a 0.40 confidence factor used for pruning, Simple NN and Complex NN with 100 training cycles, learning rate equal to 0.3, momentum to 0.2, normalized attributes and with the mean squared error cost function.

Next, we analyze the experimental results obtained for the case of population drift, population drift with non-represented cases and class definition changes.

For simplicity, we will compare the classifiers' performance using accuracy, despite the weaknesses of this measure (see [6], for example, for a discussion). The same study could be repeated using another performance measure that would focus on different trends in the classifier's behavior.

The drop in performance of a classifier deployed in a test environment will be assessed by means of our new *performance Deterioration* metric (pD), that refers

to the proportion of the difference between the trivial classifier and a classifier trained with the same data distribution used for testing. More specifically, pD is defined as

$$pD = \begin{cases} \dfrac{E_{test} - E_{ideal}}{E_0 - E_{ideal}} & \text{if } E_{test} \leq E_0 \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

where E_0 stands for the frequency of errors of the trivial classifier (which assigns data to the majority class), E_{test} is the frequency of errors on the test set and E_{ideal} is the error frequency of the classifier if it had been trained with the same distribution as the test distribution. This metric is similar to the analysis conducted in [3] to compare the relative performance between two classifiers.

When the classifier performance is no worse than the trivial classifier, it takes values in the interval [0-1]. A value close to zero indicates low deterioration, that is, that classifier performance is close the ideal situation. A value equal to one means that the classifier is no better, and, perhaps, worse than the trivial classifier in that environment.

4.1 Experiment 1: Population drift

This experiment allows us to see the performance of the classifiers trained on a Negative Growth Population (NGP) and deployed, afterwards, in different environments that may have different proportions of poor/rich, young/old people, flu symptoms, but all of them represented in the training set. Fig. 2 shows the error rate for 8 different test conditions: NGP, DP, ZGP, NGP/W, NGP/SW, NGP/P, NGP/DW, NGP/P+DW.

It can be seen that the relative performance between the classifiers does not hold under changing environments. Complex NN and Decision Trees are affected in almost the same way by changes in the data distribution. 1R shows a worse performance, but follows the same trend as Complex NN and Decision Trees, except for NGP/DW and NGP/P+DW where its performance is relatively much worse. Simple NN shows a different trend (sometimes yielding a decrease while sometimes yielding an increase in relative performance vis-à-vis the complex classifiers). It is worth highlighting, though, that under the seven changing conditions assessed here, the ranking between the classifier remains the same, even though relative differences in performance may become larger or smaller. Next, we will analyze a few cases in more detail.

Looking at the DP case in Fig. 2 may suggest that classifiers are experimenting a relevant decrease in performance with respect to the one observed in the design set NGP: the Complex NN's error rate increases from 18.6% to 24.4%, the Decision tree's from 18.9% to 24.7%, the 1R from 23.7% to 34.2% and the Simple NN from 21.2% to 33.4%. However, we may ask the question of whether this decrease in performance is due to a mismatch between the training and test conditions or whether it is only caused by the fact that the data that comes from a DP environment is more difficult to classify. Taking this eventuality into consideration, how much performance deterioration is indeed taking place?

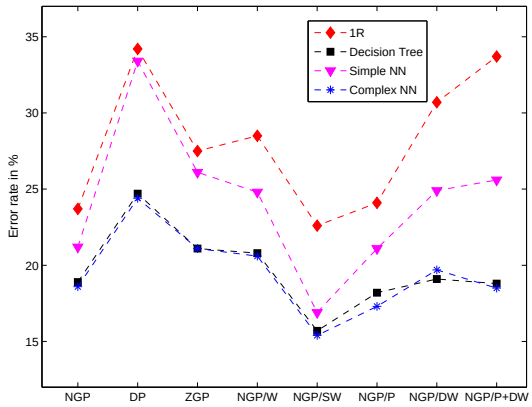


Fig. 2. Error rate (in %) for classifiers trained with data drawn from a Negative Growth Population(NGP) and deployed in different environments with population drift: DP, ZGP, NGP/W, NGP/SW, NGP/P, NGP/DW, NGP/P+DW.

Table 4. Classifier performance under a *Population drift with all the cases represented* in the training set. The error rate of a trivial classifier, an ideal situation (the same training and test set distribution), and a classifier trained with data from a NGP are shown. Performance Deterioration (pD) is shown for each particular case.

		Test Sets								
		NGP	**DP**	**ZGP**	**NGP W**	**NGP SW**	**NGP P**	**NGP DW**	**NGP P+DW**	**Averaged pD**
Trivial Classifier	Error rate	36.0	32.9	40.8	37.2	34.4	39.8	41.4	46.2	
1R	Training: **NGP**	23.7	34.2	27.5	28.5	22.6	24.1	30.7	33.7	
	Ideal situation	23.7	32.8	27.3	27.2	19.0	23.6	28.3	27.4	
	pD		1	0.01	0.13	0.24	0.03	0.19	0.33	0.28
Decision Tree	Training : **NGP**	18.9	24.7	21.1	20.8	15.7	18.2	19.1	18.8	
	Ideal situation	18.9	23.4	20.7	20.2	15.1	16.8	18.7	16.8	
	pD		0.14	0.02	0.04	0.03	0.06	0.02	0.07	0.05
Simple NN	Training: **NGP**	21.2	33.4	26.1	24.8	16.9	21.1	24.9	25.6	
	Ideal situation	21.2	25.9	25.6	23.7	16.4	20.4	23	22.9	
	pD		1	0.03	0.08	0.03	0.04	0.10	0.12	0.20
Complex NN	Training: **NGP**	18.6	24.4	21.1	20.6	15.4	17.3	19.7	18.5	
	Ideal situation	18.6	23.8	20.8	20.2	14.9	16.6	18.6	16.7	
	pD		0.07	0.02	0.02	0.03	0.03	0.05	0.06	0.04

Working with our controlled environment allows us to compute the actual performance deterioration that is taking place and understand that the analysis is not as straightforward as it may have appeared at first. Table 4 shows the error rate obtained by each classifier under different population drift scenarios (training set drawn from NGP) and under ideal conditions (training set drawn from the same distribution as the test set). In column DP of Table 4 we

see that the Decision Tree classifier deviates from 23.4% (ideal conditions) to 24.7%, indicating a performance deterioration of 0.14, while the Complex NN's deterioration is 0.07. The 1R and Simple NN performance deterioration, on the other hand, are equal to 1, meaning that they become useless in practice since their performance is worse than or equal to that of the trivial classifier. To sum up, the increase in error rate observed in our experiments is in part due to a performance deterioration and in part due to an increase in the classification problem difficulty.

On the other hand, focusing on the case NGP/SW we may think that the effort put in developing a more complex classifier may be worthless since, eventually, the Simple NN performance is very close to that of a more complex NN. But, does it mean the Simple NN is more robust under this particular change? It can be seen in Table 4 that the performance deterioration experienced by the Decision Tree, the Simple NN and the Complex NN is 0.03 for all of them. In this case, the reason why the Simple NN's error rate is very close to those of the more complex classifiers is just that the NGP/SW environment is easier to classify. Note also that the error rate is even lower than that of the original NGP.

Analyzing the seven environments with a population drift as a whole, the averaged performance deterioration is much higher for the simple models (0.28 for 1R and 0.20 for the Simple NN) than for the more complicated ones (0.05 for the Decision Tree and 0.04 for the Complex NN). To sum up and get back to the questions posed in Sect. 2, we found that: (a) a drop in performance is observed by all the classifiers but to a lesser extent by the complex ones and (b) although differences may increase or decrease when a population drift takes place, the complex classifiers remain either more accurate or as accurate as the simple ones.

4.2 Experiment 2: Population drift-NR (Not Represented cases)

In this experiment we assess the classifiers' error rate deterioration in the NGP domain when some groups (clusters) are not represented in the design set. Instances are removed based on the age attribute. As a result, we have different training data sets with: no Infants, no Teenagers, no Young Adults, no Old Adults, no Adults, no Elderly, as well as, data sets where two groups are absent.

Table 5[3] shows more severe performance deterioration for the scenarios in which the missing population has a high representation in the test set (Young Adults, Adults, Old Adults). Both complex and simple classifiers are affected by these population drifts with unrepresented cases with the simple classifiers behaving similarly to one another ($pD = 0.29$ for 1R and $pD = 0.23$ for the Simple NN) and in the complex ones too ($pD = 0.17$ for the Decision Tree and $pD = 0.18$ for the Complex NN). Note, though, that the difference in robustness against unrepresented cases for the complex classifiers when compared with the simple ones (on average pD of 0.26 against 0.175) is not as pronounced as it was in the case where all groups were represented (on average pD of 0.24 vs. 0.045).

[3] Due to space limitations, we omit a figure to represent the information in the Table.

Table 5. Classifier performance under a *Population drift with Not Represented cases* in the training set. Error rate of the trivial classifier and a classifier trained with data from a NGP population are shown as well as the performance Deterioration (pD).

		Test Set: NGP							
		1R		**Decision Tree**		**Simple NN**		**Complex NN**	
		Error	pD	Error	pD	Error	pD	Error	pD
	Trivial classifier	36.0		36.0		36.0		36.0	
Training =Test conditions		23.7		18.9		21.2		18.6	
Training set:	No Infant	23.7	0.00	19.0	0.01	22.2	0.07	19.2	0.04
	No Teenager	24.0	0.02	19.4	0.03	21.3	0.00	18.6	0.00
	No Young	25.0	0.11	18.9	0.00	23.3	0.14	19.8	0.07
	No Adult	26.5	0.23	22.2	0.19	23.5	0.16	20.2	0.09
	No Old Adult	24.6	0.07	19.6	0.04	23.8	0.17	19.8	0.07
	No Elderly	24.1	0.03	19.5	0.03	22.9	0.11	19.3	0.04
	No Old Adult+No Elderly	33.3	0.78	24.5	0.33	32.7	0.77	27.4	0.50
	No Infant + No Teenager	24.4	0.06	20.7	0.11	21.4	0.01	20.0	0.08
	No Elderly + No Infant	24.2	0.04	20.0	0.06	21.4	0.01	19.2	0.04
	No Old Adult+No Teenager	25.0	0.11	20.2	0.08	21.8	0.04	19.2	0.04
	Averaged pD		0.29		0.17		0.23		0.18

In summary, (a) all classifiers show a drop in performance when there are unrepresented cases in the training set, with (b) a slightly higher impact on simple classifiers than in complex ones.

4.3 Experiment 3: Class definition changes.

Error rates for the four classifiers have also been assessed in the presence of changes in class definition: rules that lead to more complications (MC) and rules that lead to fewer complications (FC). A comprehensive evaluation has been carried out assessing all the possible mismatch combinations (see Table 6 for details). It turns out that the Simple NN, the Complex NN and the Decision Trees are affected on average in the same way (pD around 0.24) by these changes in class definition. The 1R classifier, however, shows more sensitivity to these changes, with its pD always higher than that of the other three classifiers, being trivial in one case and with an overall pD equal to 0.36. In [3], Hand suggests that under class definition changes it is possible that models that fit the training data less well will perform better in the new changing environments. Our analysis point out that in this experimental framework the simple 1R rule is affected to a greater extend than the other three classifiers for the class definition change defined here. On the other hand, we have not found significant differences between the simple NN and the other complex classifiers in terms of performance deterioration. Such a result, thus, reflects Hand's suggestion, except for the fact that the complex classifiers, that yielded better results in the original scenario, will also yield better classification rules under the class definition changes evaluated here, and, thus, remain a better choice.

Table 6. Class Definition Changes. Error rate of the trivial classifier is shown together with the error rate for a classifier trained and tested with different sets: original rules (NGP), fewer (FC) and more (MC) complications. Performance Deterioration (pD) is shown (The symbol - is used when no deterioration performance applies).

		Test: NGP		**Test : MC**		**Test: FC**		**Averaged**
		Error	pD	Error	pD	Error	pD	**pD**
Trivial classifier	**Training conditions**	36.0		28.4		34.4		
1R	**NGP**	23.7	-	27.1	0.33	22.8	0.10	
	Training MC	26.1	0.20	26.5	-	26.1	0.36	
	FC	25.8	0.17	29.5	1	21.5	-	
								0.36
Decision Tree	**NGP**	18.6	-	21.6	0.08	19.0	0.08	
	Training MC	20.6	0.11	21.0	-	24.1	0.38	
	FC	20.4	0.11	25.9	0.67	17.7	-	
								0.24
Simple NN	**NGP**	21.2	-	26.3	0.17	19.4	0.08	
	Training MC	23.1	0.13	25.9	-	23.0	0.30	
	FC	22.0	0.05	25.9	0.73	18.1	-	
								0.24
Complex NN	**NGP**	18.6	-	21.6	0.12	19.0	0.10	
	Training MC	20.8	0.13	20.7	-	23.4	0.36	
	FC	20.5	0.11	25.0	0.56	17.3	-	
								0.23

5 Conclusions and further research

In this work we have evaluated the impact of several changing environments on simple classifiers (1R, Simple NN) and more sophisticated ones (Complex NN, C4.5 Decision Tree). Our results show that the trend noticed by David Hand [3] does happen in some cases, while in other cases, differences between simple and sophisticated classifiers become wider under changing conditions.

By analyzing the behavior of two different classes of classifiers in an artificial environment, we have observed that the resultant relative performance between the simple and complex classifiers under changing conditions can be decomposed as the additive effect of the classifier's performance deterioration and the complexity of the new classification scenario. Thus, a decrease in the error rate difference between a complex and a simple classifier may be caused by the fact that (i) the complex classifier (with lower initial error rate) is more vulnerable to changing conditions and/or (ii) the classification problem becomes easier for the simple classifier. The same reasoning applies when the difference increases.

Given our use of a controlled classification framework, we were able to propose a measure of performance deterioration which compares the results obtained in a changed environment to those obtained in an ideal situation with no mismatch. Our experimental results show that the changing conditions evaluated here lead to a drop in performance by all classifiers. In the case of *population*

drift it is higher for simple classifiers than for the more sophisticated ones, while in the case of *population drift with non-represented cases* the differences between the two categories of classifiers become less pronounced. Finally, under *class definition changes*, putting aside the simple 1R rule (which was shown to be very sensitive to all the shifts evaluated here), the remaining three classifiers show the same amount of performance deterioration. Results in this experimental framework, thus, show that simple classifiers do not become more accurate than complex ones, although we can not expect the differences in error rates to hold under changing environments. They may increase or decrease depending on the kind of changes that take place in the data.

Our immediate future work will consist of studying the effect of training classifiers on different data sets of various sizes. Longer term studies will include the development of more realistic models for artificial data set generation.

References

1. R. Alaiz-Rodríguez, A. Guerrero-Curieses, and J. Cid-Sueiro. Minimax regret classifier for imprecise class distributions. *Journal of Machine Learning Research*, 2007.
2. C. Drummond and R. C. Holte. Cost curves: An improved method for visualizing classifier performance. *Machine Learning*, 65(1):95–130, 2006.
3. D.J. Hand. Classifier technology and the illusion of progress. *Statistical Sciences*, 21(1):1–15, Jun 2006.
4. R. Holte. Elaboration on two points raised in "classifier technology and the illusion of progress". *Statistical Science*, 21(1), 2006.
5. J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 601–608. MIT Press, Cambridge, MA, 2007.
6. N. Japkowicz. Why question machine learning evaluation methods?. an illustrative review of the shortcomings of current methods. In *AAAI-2006 Workshop on Evaluation Methods for Machine Learning*, pages –, Boston, USA, 2006.
7. M. G. Kelly, D. J. Hand, and N. M. Adams. The impact of changing populations on classifier performance. In *Proceedings of Fifth International Conference on SIG Knowledge Discovery and Data Mining*, pages 367–371, San Diego, CA, 1999.
8. T. Lane and C. E. Brodley. Approaches to online learning and concept drift for user identification in computer security. In *Knowledge Discovery and Data Mining*, pages 259–263, 1998.
9. F. Provost and T. Fawcett. Robust classification systems for imprecise environments. *Machine Learning*, 42(3):203–231, March 2001.
10. M. Saerens, P. Latinne, and C. Decaestecker. Adjusting a classifier for new a priori probabilities: A simple procedure. *Neural Computation*, 14:21–41, January 2002.
11. H. Shimodaira. Improving predictive inference under convariance shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 2000.
12. I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, October 1999.
13. K. Yamazaki, M. Kawanabe, S. Watanabe, M. Sugiyama, and K. Müller. Asymptotic bayesian generalization error when training and test distributions are different. In *ICML 2007*, pages 1079–1086, New York, NY, USA, 2007. ACM Press.