Viability of an Alarm Predictor for Coffee Rust Disease Using Interval Regression*

Oscar Luaces[1], Luiz Henrique A. Rodrigues[2], Carlos Alberto Alves Meira[3], José R. Quevedo[1], Antonio Bahamonde[1]

[1] Artificial Intelligence Center. Universidad de Oviedo at Gijón, Asturias, Spain
www.aic.uniovi.es
[2] Universidade Estadual de Campinas, Faculdade de Engenharia Agrícola, Cx. P. 6011, CEP 13083-875 Campinas, SP, Brazil
[3] Embrapa Informática Agropecuária, Cx. P. 6041, CEP 13083-970 Campinas, SP, Brazil

Abstract. We present a method to formulate predictions regarding continuous variables using regressors able to predict intervals rather than single points. They can be learned explicitly using the so-called insensitive zone of regression Support Vector Machines (SVM). The motivation for this research is the study of a real case; we discuss the feasibility of an alarm system for coffee rust, the main coffee crop disease in the world. The objective is to predict whether the percentage of infected coffee leaves (the incidence of the disease) will be above a given threshold. The requirements of such a system include avoiding false negatives, seeing as these would lead to not preventing the disease. The aim of reliable predictions, on the other hand, is to use chemical prevention of the disease only when necessary in order to obtain healthier products and reductions in costs and environmental impact. Although the breadth of the predicted intervals improves the reliability of predictions, it also increases the number of uncertain situations, i.e. those whose predictions include incidences both below and above the threshold. These cases would require deeper analysis. Our conclusion is that it is possible to reach a trade-off that makes the implementation of an alarm system for coffee rust disease feasible.

1 Introduction

In this paper we discuss how to learn alarm functions in a real world problem. Starting from a faithful description of present circumstances, these functions must predict future situations of risk so that we may then act to prevent any foreseeable damage. In this context, the costs of prediction errors are not symmetrical: the consequences of false prediction of an alarm (*false positives*) are often not as serious as those of predicting false non-alarms (*false negatives*).

* The research reported here is supported in part under grant TIN2008-06247 from the MICINN (Ministerio de Ciencia e Innovación, of Spain), and grant 2009/07366-5 from the FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo, Brazil)

We deal with continuous target variables whose values above a given threshold should be notified as soon as possible with the highest degree of accuracy. We try, at least, to minimize the percentage of false negatives. The straightforward approach is to learn a regressor: in addition to providing alarm alerts, the regressor produces a numeric assessment of how serious the situation may be.

We present an agriculture case study. The incidence of coffee rust epidemics is caused by a fungus called *Hemileia vastatrix* Berk. & Br., a devastating disease to coffee plantations. This incidence can be measured by the percentage of leaves infected by the fungus. It is well known that the factors that stimulate the growth of the fungi are weather conditions, the type of plantation and the current incidence. Thus, a regression learning task must include these features as predictors and the future incidence as the target value.

We trained a Regression Support Vector Machine (*SVM*) with quite good results. The correlation between predicted and actual incidences is about 0.94 in a cross-validation experiment. However, if we try to devise an alarm system for predicting values above a given threshold, we find that the number of false negatives is too high. In this case the threshold is $\tau = 4.5$. This is not an academic parameter, it is the threshold used in Brazilian plantations; see [6–8] for a detailed discussion.

To overcome this weakness of regression, we try to learn models to predict *approximations* to incidence values instead of exact values. To implement this idea, there are a number of possible alternatives. In the approach presented here, we relax the specifications of regression, changing target points for intervals of a fixed width, say 2ϵ. Following [1, 5], these predictors may be called *nondeterministic* regressors.

The method employed to learn intervals of fixed width uses regression SVM. These learning algorithms search for predictors that minimize a loss function which ignores errors situated within a certain distance (ϵ) of the true value: ϵ-*insensitive* loss functions.

To transform interval predictions into alarms, we adopt a cautious policy. Only those intervals completely included below the threshold will be understood as non-alarms. On the other hand, if a predicted interval is above the threshold, that will mean an alarm. However, we have a third possibility situated somewhere in between: predicted intervals that include points above and below the threshold. We label these situations as *warnings*. The usefulness of warnings is that they capture classification errors of pure deterministic regressors. In fact, these errors arise for predictions near the threshold.

In other words, we can convert misclassifications into a type of situation that may require deeper analysis. However, when the alarm system predicts an alarm, and especially a non-alarm, the confidence in these predictions is very high. The radius ϵ of the intervals is proportional to the number of warnings and hence to the prudence of the whole alarm system. In this sense, our approach is closely related to that of classifiers with a reject option [2, 4].

Our conclusion is that a trade off between the number of false negatives and warnings would lead to a useful alarm system for coffee rust. The search for an optimal value for ϵ is beyond the scope of this paper, as we would have to consider the important economic and environmental aspects involved in coffee growing. Nonetheless, considering the results reported at the end of the paper (Section 6), the feasibility of implementing an alarm system is guaranteed. Moreover, the only requirement is a cheap weather station.

In the next section the coffee rust disease and the dataset used in the paper are presented in detail. Sections 3 and 4 are devoted to deterministic and nondeterministic alarms respectively. In Section 5 we discuss the temporal perspective of the alarms.

2 Coffee Rust

The coffee rust caused by fungi *Hemileia vastatrix* Berk. & Br. is the main coffee crop disease in the world. In Brazil, damages lead to yield reduction of up to 35% in regions where climate conditions are propitious to the disease. The impact is thus considerable due to the economic importance of coffee crop.

The traditional way to prevent the disease is to apply agrochemical fungicides on fixed calendar dates. However, the fungicides contaminate the environment and reduce the quality of the coffee. Moreover, as the intensity of the disease between seasons suffers major variations, the use of agrochemicals is not always justified.

The aim of this paper is to discuss the viability of building alarm functions to alert on high incidences of coffee rust. The purpose would be to build economically viable control measures. Our proposal is a predictor, learned using data mining tools, that would allow applying agrochemicals only when necessary, leading to healthier products and reductions in costs and environmental impact.

It is important to emphasize here that fungicides must by applied in advance since they need several days to take effect in coffee plants. Having all this in mind, we used a dataset [6–8] whose temporal dimension is very important. The data was obtained on a monthly basis from an experimental farm (Fundação Procafé, Varginha, MG, Brazil), from October 1998 to October 2006, with reports of coffee rust incidences. In September of each year (beginning of agricultural season), eight plots producing coffee were selected, four with thin spacing (approximately 4000 plants/ha) and four with dense spacing (approximately 8000 plants/ha). For each case, two plots were selected with high fruit load (above 1800 kg/ha) and two with low fruit load (below 600 kg/ha). There was no disease control in those plots. Meteorological data was automatically registered every 30 minutes by a weather station close to where the incidence of coffee was being evaluated.

2.1 The Learning Task

From a formal point of view, throughout this paper we deal with the dataset described as follows.

Let $\mathcal{X}$ be a set of descriptions of current situations. Here we wanted to represent, using the data collected, the idea that an alarm system can be used at any time, not only from the first day of one month to guess the incidence in the first day of the next month. In the coffee rust problem, if we want to predict the incidence of the fungi in a *target* day, we consider predictions made with different days ahead. Thus $\mathcal{X}$ is a set of vectors whose components are:

- Fruit load of the plantation: low (1) or high (2)
- Spacing between plants: dense (1) or thin (2)
- Percentage of leaves infected by fungi in date d_0
- Days from d_0 till now (the day we make the prediction)
- Days from now till the target day: 1 month, 25, 20, 15 and 10 days
- Weather scores in the last 45 days

The weather scores are 13 variables per day, and they include: temperatures, solar radiation, number of hours with sun light, wind speeds, rain, relative humidity, number of hours with relative humidity above 95%, average temperature during these hours, and the same values but during the night. For more details, see [7, 8].

Therefore, the dimension of the vectors of the input space $\mathcal{X}$ is 590. On the other hand, the output space in this case is just the interval of real numbers, $\mathcal{Y} = [0, 100]$, to capture the percentage of coffee leaves inflected by the fungi.

3 Regression and Deterministic Alarms

We start presenting the baseline approach obtained from a standard regression tool. From a formal point of view, learning tasks can be presented in the following general framework. Let $\mathcal{X}$ be an input space, and let $\mathcal{Y}$ be an output space. A *learning task* is given by a training set $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$ drawn from an unknown distribution $Pr(X, Y)$ from the product $\mathcal{X} \times \mathcal{Y}$. The aim of such a task is to find a hypothesis h (of a space $\mathcal{H}$ of functions from $\mathcal{X}$ to $\mathcal{Y}$) that optimizes the *expected prediction performance (or risk)* on samples independently and identically distributed (i.i.d.) according to the distribution $Pr(X, Y)$:

$$R^{\Delta}(h) = \int \Delta(h(\boldsymbol{x}), y) \; d(Pr(\boldsymbol{x}, y)), \tag{1}$$

where $\Delta(h(\boldsymbol{x}), y)$ is a loss function that measures the penalty due to the prediction $h(\boldsymbol{x})$ when the true value is y.

If $\mathcal{Y}$ is a metric space (usually the set of real numbers), the learning job is a *regression* task, as in the case of coffee rust. In this case, the aim of learners is to obtain a hypothesis whose predictions are as similar as possible to actual values in the output space. This can be accomplished, for instance, using least squares regression.

On the other hand, the goal of SVM regressors is to minimize the so-called ϵ-*insensitive* loss function. If ϵ is a positive value, this loss does not penalize predictions whose distance to true values is below ϵ; in symbols,

$$\Delta_{\epsilon}(h(\boldsymbol{x}), y) = \max\{0, |h(\boldsymbol{x}) - y| - \epsilon\}. \tag{2}$$

In any case, once we have learned a regressor h, if τ is a threshold in $\mathcal{Y}$, we interpret the outputs of h as follows

$$Alarm(h(\boldsymbol{x})) = \begin{cases} \text{non-alarm} & h(\boldsymbol{x}) \in (-\infty, \tau] \\ \text{alarm} & h(\boldsymbol{x}) \in (\tau, +\infty). \end{cases} \tag{3}$$

Notice that the performance of what has been learned can be measured in two different but complementary ways: using the scores of regressors applied to h, and the scores of classifiers applied to $Alarm \circ h$.

4 Regression with Broad Insensitive Zone: Nondeterministic Alarms

Let us assume that we have a regressor whose accuracy to predict a continuous variable is not completely satisfactory. For instance, the performance of a regressor may fail when it is measured in terms of alarm classifications (Eq. 3). This is the case of regressors obtained from the coffee rust learning task. The scores will be discussed later in Section 6. To overcome this problem, as was explained in the introduction, we are going to use regressors allowed to predict intervals rather than single points.

The idea is that the true class of an entry $\boldsymbol{x}$ may be *somewhere* into the predicted interval for $\boldsymbol{x}$. A simple way to implement this idea is to look for regressors that predict intervals of a fixed width, say 2ϵ. Notice that this is exactly the semantics of ϵ-insensitive zone (Eq. 2). For later reference, we recall the formulas of SVM regressors here.

Given a regression learning task S (Section 3) and a *tube* value $\epsilon > 0$, a regression SVM learns a function

$$h_\epsilon(\boldsymbol{x}) = \sum_{i=1}^{n} (\alpha_i^- - \alpha_i^+) K(\boldsymbol{x}_i, \boldsymbol{x}) + b^*, \tag{4}$$

where K is the *rbf* kernel, $K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$; b^*, and $\boldsymbol{\alpha}^+$, $\boldsymbol{\alpha}^-$ are respectively the solution and the Lagrange multipliers of the following convex optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w}, b, \xi} \quad & \frac{1}{2}\langle \boldsymbol{w}, \boldsymbol{w} \rangle + C \sum_{i=1}^{n} (\xi_i^+ + \xi_i^-), \\ \text{s.t.} \quad & (\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i) \rangle + b) - y_i \leq \epsilon + \xi_i^+, \quad y_i - (\langle \boldsymbol{w}, \phi(\boldsymbol{x}_i) \rangle + b) \leq \epsilon + \xi_i^-, \\ & \xi_i^+, \xi_i^- \geq 0, \quad i = 1, \ldots, n. \end{aligned} \tag{5}$$

The interval regressor associated to h_ϵ is then defined by

$$h_{ND(\epsilon)}(\boldsymbol{x}) = [h_\epsilon(\boldsymbol{x}) - \epsilon, h_\epsilon(\boldsymbol{x}) + \epsilon]. \tag{6}$$

Notice that we have one optimal interval regressor for each value of the tube, ϵ. The regressor h_ϵ, accordingly to (Eq. 5) is different for each value of ϵ. When the aim is to learn a deterministic regressor, typically ϵ is a small number; by default, we use $\epsilon = 0.1$. However, for interval predictions, we may use wider tubes.

However, the problem of interval regressors is that they are not as precise as regular regressors. There is some degree of vagueness in interval predictions. For this reason, following [1, 5], we call them nondeterministic predictors.

To handle this type of predictions, we have to reformulate the alarms associated with a regressor. We need to interpret predictions that include, at the same time, alarms and non-alarms. Our proposal is to label these situations as *warnings*: something between alarms and non-alarms. The scores reported in Section 6 will illustrate the advantages of nondeterministic alarm functions.

Formally, we propose the following extension of (Eq. 3). Given an interval regressor h_{ND}, if τ is a threshold in $\mathcal{Y}$, we shall interpret the outputs of the regressor as follows

$$Alarm(h_{ND}(\boldsymbol{x})) = \begin{cases} \text{non-alarm} & h_{ND}(\boldsymbol{x}) \subset (-\infty, \tau] \\ \text{alarm} & h_{ND}(\boldsymbol{x}) \subset (\tau, +\infty) \\ \text{warning} & otherwise. \end{cases} \tag{7}$$

Additionally, from the classification point of view, given a nondeterministic regressor, h_{ND}, for a test set $S' = \{(\boldsymbol{x'}_1, y'_1), \ldots, (\boldsymbol{x'}_m, y'_m)\}$, it is important to measure the proportion of test examples that fall outside the tube

$$out\_tube(h_{ND}, S') = \frac{1}{m} \sum_{i=1}^{m} 1 - \left(y'_i \in h_{ND}(\boldsymbol{x'_i})\right). \tag{8}$$

5 Time Series Alarms

As was explained in Section 2.1, an alarm system for coffee rust would be able to be used at any time. To simulate this capacity, given an incidence percentage y measured the first day of one month, we considered different values $t \in [30, 25, 20, 15, 10]$ for the number of days ahead of predictions. For each t we have the corresponding weather records. Thus, in the learning task, for each y we have a time series

$$\{(\boldsymbol{x}_t, y) : t \in [30, 25, 20, 15, 10]\}. \tag{9}$$

To evaluate the sequence of alarm alerts produced by an interval regressor h_{ND} in (Eq. 9), the idea is that if an alarming prediction occurs for some t, then the reaction would be to use the agrochemical fungicides; any subsequent notice of non-alarm would not be heard. On other hand truly non-alarming predictions for y would need a sequence of non-alarms for all t values. Formally, this point of view is captured by the following definition

$$Alarm(h_{ND}(\boldsymbol{x}_t)) = \begin{cases} \text{non-alarm} & \forall t, h_{ND}(\boldsymbol{x}_t) \subset (-\infty, \tau] \\ \text{alarm} & \exists t, h_{ND}(\boldsymbol{x}_t) \subset (\tau, +\infty) \\ \text{warning} & otherwise. \end{cases} \tag{10}$$

Table 1. Regression scores obtained (using cross-validation) for different values of the radius ϵ of the insensitive zone or *tube*. In rows, for each combination of fruit load (l) and spacing (s), we report the averages of *absolute error*, ϵ-insensitive loss, Δ_ϵ (Eq. 2), and *correlations*. The last rows shows the scores considering at the same time all types of plantations

	Score	$\epsilon = 0.1$	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 3$	$\epsilon = 4$
$l = 1$	*absolute error*	6.53	6.11	5.86	5.75	5.72
$s = 1$	Δ_ϵ	6.53	6.02	5.62	5.23	4.79
	correlation	0.81	0.82	0.83	0.84	0.84
	out_tube	0.98	0.80	0.73	0.65	0.54
$l = 1$	*absolute error*	7.40	7.07	6.87	6.88	7.13
$s = 2$	Δ_ϵ	7.40	7.00	6.60	6.35	6.41
	correlation	0.82	0.82	0.82	0.82	0.82
	out_tube	1.00	0.86	0.73	0.63	0.61
$l = 2$	*absolute error*	6.45	6.33	6.20	6.14	6.29
$s = 1$	Δ_ϵ	6.45	6.27	5.96	5.71	5.40
	correlation	0.96	0.96	0.96	0.96	0.96
	out_tube	0.98	0.88	0.76	0.66	0.55
$l = 2$	*absolute error*	6.56	6.13	5.84	5.73	5.80
$s = 2$	Δ_ϵ	6.56	6.05	5.59	5.16	4.90
	correlation	0.96	0.96	0.97	0.97	0.97
	out_tube	0.99	0.83	0.75	0.64	0.56
all	*absolute error*	6.74	6.41	6.19	6.12	6.23
	Δ_ϵ	6.73	6.34	5.94	5.61	5.38
	correlation	0.94	0.94	0.94	0.94	0.95
	out_tube	0.98	0.84	0.74	0.64	0.57

6 Experimental Results

In this section we report a number of experiments carried out to illustrate the role played by the width of intervals involved in alarm predictions. With the dataset introduced in Section 2.1, we used a 10-fold cross validation to estimate the scores reported in the following figure and tables. As was mentioned in the introduction, in all the experiments, the threshold used to discriminate alarms was $\tau = 4.5$.

The SVM regressors were learned using LibSVM [3], with an rbf kernel. The parameters C and σ were adjusted using an *internal grid search* in each training set. The ranges for this search were: $C \in [0.001, 0.01, 0.1, 1, 10, 100, 1000]$, and $\sigma \in [0.01, 0.1, 0.3, 0.5, 0.7]$. The search employed an internal 2-fold cross validation repeated 3 times; the aim being to optimize the average Δ_ϵ (Eq. 2).

First we compared the scores achieved from the point of view of regression for different values of ϵ and different plantation types. The results are gathered in Table 1.

We observe that correlations are quite different from the data of plantations with low ($l = 1$) and high ($l = 2$) fruit loads. The quality of regressors is worse in the case of low fruit load. However, the correlation obtained for the whole dataset

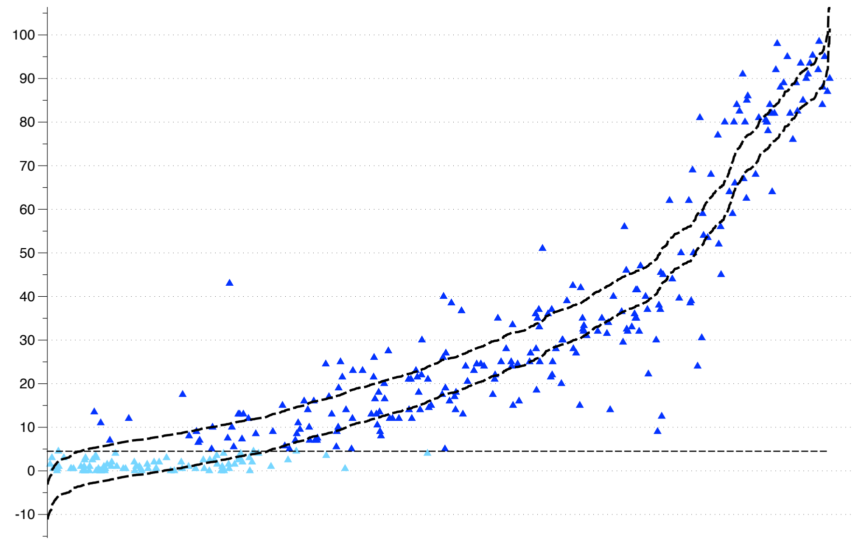

Fig. 1. True incidence percentages (▲) and the predicted intervals by a regressor $h_{ND(4)}$. The horizontal axis represents the indexes of samples ordered according to their predictions. We only included predictions made one month in advance to make the figure more clear. The proportion of points outside the tube are similar if we vary the days ahead of predictions. The horizontal dashed line represents the threshold $\tau = 4.5$

is quite high, around 0.94. The value of the radius of the predicted interval, ϵ, has no influence on these results. But, of course, ϵ has a dramatic impact in the proportion of points outside the tube. Here, the results range from almost all to 0.57. Obviously, it is easier to include examples inside wider tubes.

In Figure 1 we represent graphically the predictions and true values. To make the figure more clear, we show only a subset of examples: predictions made one month ahead. We used the predictions of the interval regressor learned with $\epsilon = 4$, $h_{ND(4)}$; that is, a regressor whose predictions are intervals with a width of 8. We can appreciate that the errors are higher when predictions range from 30 to 50.

Time series. In Table 2 we report the results obtained by the alarm functions obtained for the time series described in Section 5. In this case, in cross-validations we took care that time series (Eq. 9) were never separated into train and test splits.

The table shows the confusion matrices obtained in cross-validations. For *deterministic* regression, the default value of the insensitive zone used was $\epsilon = 0.1$. In this case, of course, there are no doubtful classifications: no warnings appear in the corresponding columns of Table 2. Unfortunately, the consequence is that the number of errors is too high: 14 false non-alarms, and 16 false alarms.

Table 2. Confusion matrices obtained (using cross-validation over time series) for different values of the radius ϵ of the insensitive zone or *tube*. Columns represent true classes: alarm (a), non-alarm ($\neg a$). Rows report the occurrences of each possible prediction *(Pre)* (alarm, warning (w), non-alarm) for each combination of load *(l)* and spacing *(s)*. The last rows shows the scores considering at the same time all types of plantations; that is, the sum of the corresponding confusion matrices

		$\epsilon = 0.1$		$\epsilon = 1$		$\epsilon = 2$		$\epsilon = 3$		$\epsilon = 4$	
	Pre	$\neg a$	a	$\neg a$	a	$\neg a$	a	$\neg a$	a	$\neg a$	a
$l = 1$	$\neg a$	18	6	15	3	12	0	3	0	1	0
$s = 1$	w	0	0	3	4	7	9	15	9	18	9
	a	5	56	5	55	4	53	5	53	4	53
$l = 1$	$\neg a$	18	6	12	2	6	0	1	0	1	0
$s = 2$	w	0	0	6	4	10	6	15	2	17	5
	a	5	56	5	56	7	56	7	60	5	57
$l = 2$	$\neg a$	16	0	14	0	11	0	3	0	2	0
$s = 1$	w	0	0	2	0	5	0	13	0	14	0
	a	4	65	4	65	4	65	4	65	4	65
$l = 2$	$\neg a$	17	2	14	0	5	0	3	0	1	0
$s = 2$	w	0	0	2	1	12	0	13	0	16	0
	a	2	64	3	65	2	66	3	66	2	66
sum	$\neg a$	69	14	55	5	34	0	10	0	5	0
	w	0	0	13	9	34	15	56	11	65	14
	a	16	241	17	241	17	240	19	244	15	241

If we use wider predicted intervals ($\epsilon \geq 1$), the number of errors decreases dramatically, but the price is that there is an increase in the number of warning predictions. Thus, for $\epsilon = 1$ the number of false non-alarms is only 5 with 22 warnings (6.5% of all cases). Let us remark that all these false non-alarms are due to plantations with low fruit load ($l = 1$), which is coherent with the results obtained for regression scores, see Table 1. With $\epsilon \geq 2$, the number of false non-alarms is zero, but the warnings rise to $14.4\%, 19.7\%$ and 23.2% respectively for $\epsilon = 2, 3, 4$.

7 Conclusion

We discussed the viability of an alarm system for coffee rust, the main coffee crop disease in the world. In this case, the aim is to apply the chemical prevention of the diseases only when necessary to achieve healthier products and reductions in cost and environmental impact. But we must be vigilant to avoid false non-alarms since they would conduct to not prevent an awful increase in the incidence of the disease.

The approach presented here proposes to handle predictions about continuous variables by regressors able to predict intervals rather than single points. They can be learned from regression learning tasks using the so-called ϵ-insensitive zone (ϵ is the radius of the predicted intervals). An optimal solution can be obtained by Regression Support Vector Machines.

The use of interval predictors allow us to distinguish a third type of situations placed between alarms and non-alarms. We called them warnings. Roughly speaking, we found that the confidence in non-alarm predictions is higher as ϵ increases, while it is quite stable for alarm predictions. Somehow, the alarm predictor becomes more prudent, but requires more frequently deeper analysis to decide what to do in uncertain (warning) situations.

A trade off between the number of non-alarms and warnings would lead to a useful alarm system for the coffee rust. If we want to search for an optimal value for ϵ, we must consider the important economic and environmental aspects involved in coffee growing.

Finally, it is worth noting here that the cost of implementing the alarm systems presented in this paper is very low. The only requirement is a cheap weather station able to register the data described in Section 2.1.

Acknowledgements

The authors are grateful to the Brazilian Fundação Pró Café for providing the data used in this paper.

References

1. Alonso, J., del Coz, J.J., Díez, J., Luaces, O., Bahamonde, A.: Learning to predict one or more ranks in ordinal regression tasks. In: Daelemans, W., Goethals, B., Morik, K. (eds.) Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD). pp. 39–54. No. LNAI 5211, Springer (2008)
2. Bartlett, P., Wegkamp, M.: Classification with a reject option using a hinge loss. Journal of Machine Learning Research 9, 1823–1840 (2008)
3. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm
4. Chow, C.: On optimum recognition error and reject tradeoff. IEEE Transactions on Information Theory 16(1), 41–46 (1970)
5. del Coz, J.J., Díez, J., Bahamonde, A.: Learning nondeterministic classifiers. Journal of Machine Learning Research 10, 2273–2293 (Oct 2009)
6. Japiassú, L., Garcia, A., Miguel, A., Carvalho, C., Ferreira, R., Padilha, L., Matiello, J.: Influência da carga pendente, do espaçamento e de fatores climáticos no desenvolvimento da ferrugem do cafeeiro. In: Simpósio de Pesquisa dos Cafés do Brasil. Águas de Lindóia, SP, Brasil (2007)
7. Meira, C., Rodrigues, L., de Moraes, S.: Análise da epidemia da ferrugem do cafeeiro com árvore de decisão. Tropical Plant Pathology 33(2), 114–124 (2008)
8. Meira, C., Rodrigues, L., de Moraes, S.: Modelos de alerta para o controle da ferrugem-do-cafeeiro em lavouras com alta carga pendente. Pesq. agropec. bras 44(3), 233–242 (2009)