Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Towards an Online, Non-stochastic Approach to Fault Detection

K. Pelckmans *

* *Syscon, Information Technology, Uppsala University, 75501, SE*

Abstract: This note considers the problem of learning a warning system from observations of a system which has not encountered any error as yet. That is, can we infer a warning rule from a system remaining in normal operation regime, without making any (stochastic) assumptions? While this problem appears to be paradoxical, methods which were studied in online learning theory can be used to find such a strategy. To illustrate that this is indeed possible, the classical linear PERCEPTRON rule is reinterpretated.

The design of methods for detection of faults has been studied intensively over at least 4 decades, see e.g. the survey of Basseville [1988]. Firstly, consider the setup where a model for both nominal behaviour as well as atypical ones is *given*. If these models are *deterministic*, then the issue of determining wether a new case belongs to either situation is direct. If the setup is *stochastic*, then determining to which situation a new case belongs is essentially answered by statistical decision theory. In practice, these boil down to performing a statistical test, and deciding wether the resulting level is significant enough for the application at hand. In many cases, one resorts to the Uniformly Most Powerful (UMP) Likelihood-Ratio (LR) tests, or a variation on this theme (see e.g. Kay [1998], Van Trees [2004]).

Note that a different approach uses the device of p-values. When only the distribution of the nominal case (the null-distribution) is specified, a p-value indicates how much a new case deviates from this model. The LR test however requires specification of both the distributions characterising the NOMINAL, as well as the distribution governing the ALARM cases. The payoff of the latter is that it comes with a sounder theoretical support (e.g. the Neyman-Pearson lemma). Both schools are still prevalent in present day (see e.g. the discussion in Lehmann [1993]).

Let's refer to this step as the *decision task*.

The subsequent question is then how to *learn or infer* the models for either NOMINAL and FAULTY cases. There exist a number of essentially different approaches for doing so:

- (Parameteric) If a model is specified up to a few parameters, for the normal behaviour of the signals, then it is not too difficult to find out when a new measurement does not follow it close enough. Such *atypical* points are obvious candidates for faulty behaviour. That is, faulty measurements follow the model *dual* to the *nominal* model. Note that in case only a few parameters need to be estimated, this can be done using NOMINAL observations following the postulated model.
- (Non-parametric) If such a model cannot be specified, or is represented using a large number of unknowns, then one has to make different assumptions to separate *nominal* behaviour from *faulty* behaviour. A common one is to make I.I.D. assumptions on the data: that is assume that the data is sampled independently from an identical distribution. The devise of mixing (see e.g. Vidyasagar [2002] for a survey) relaxes this condition to dependent data, typically found in a control or signal processing setting. This case is also referred to as *non-parametric*, see e.g. Lehmann [2006].
- (Classification) There is a different line of research for addressing this task. Let nominal measurements be labeled as $y_t = -1$, and faulty ones have a $y_t = +1$ label. Then, based on the measurements and their corresponding labels, one tries to find an optimal rule which separate this two classes of examples. This line of thinking is often followed in a setting of machine learning algorithms, see e.g. Mohri et al. [2012], Hastie et al. [2001]. Most of those approaches need customisation when handling imbalances of the classes. In the ALARM setting, there is a natural imbalance as ALARMS are by nature much less frequent than NOMINAL behaviour. In fact, we are looking to the extreme case where one can only observe one type of data. Hereto, so-called one-class classification methods were proposed as a technique to reduce such problem to a traditional classification problem. The principal thinking again is to build the *least complex* model explaining the NOMINAL data. In case of Support Vector Machines (SVMs) (see e.g. Hastie et al. [2001]), complexity is interpreted in terms of margin, leading to the one-class Support Vector Machine, see e.g. Shawe-Taylor and Cristianini [2004].. However it was observed that this approach is in practice no viable alternative to the parametric or non-parameteric approaches, while often requiring much heavier computations and careful tuning.

Let's refer to this step as the *learning task*.

This paper explores a new approach integrating both the *decision task*, and the *learning task*. Specifically, we will phrase the problem as an *online learning problem*, and

⋆ This work was supported in part by Swedish Research Council under contract 621-2007-6364.

point out the resulting advantages. The key idea is to update the model whenever it makes a mistake. That is, when it predicts an ALARM while there was none, the learner is asked to update its rule. Such strategy is by now quite standard in the area of theoretical machine learning (Cesa-Bianchi and Lugosi [2006], Mohri et al. [2012]), but it is still highly non-conventional in a context of identification and automatic control. The surprising bit of the consequent theory is that *no stochastic assumptions* need to be made in order to give formal results.

This ideas find a natural application in a systems and control setting as follows. Successful control applications demand a proper monitoring system. However, since the data is generated by complex feedback loops, stochastic assumptions as *independent sampling* on the involved signals are often problematic. It is often desirable to work with techniques which work well with arbitrary signals, that is, in a $\|\cdot\|_\infty$ sense. The theory of online learning provides such a framework, and the connection with H_∞ is studied in Hassibi et al. [1996]. A main open question however is how to handle effectively the fault detection case, and whether are there opportunities where the non-stochastic nature of the online learning schemes can enrage insights in control and recursive identification? This then provides the motivation for the work in this paper. Further integration of this line of thinking within engineering applications as in Gertler [1998] is left as future work.

This paper is organised as follows. The next section formalises the setup and gives the form of the answer. Section 1 states the basic result and gives a proof. The first subsection examines how one can extend the PERCEPTRON rule to handle the fault detection case, while the second subsection gives a way to improve the technique using the devise of nuclear norms. Section 2 provides numerical examples, and Section 3 concludes the paper.

1. ALARM RULE

Protocol of learning from a passive teacher:

Given f_0
FOR $t = 1, 2, \ldots$
(1): A measurement x_t comes in.
(2): The learner issues ALARM or NOT, based on x_t and f_{t-1}.
(3): In case of a conjectured ALARM, the teacher checks.
(4): The learner adjusts $f_{t-1} \to f_t$ based on feedback (FP or TP).
END

Table 1. The protocol for learning a detection rule. The aim of the learner is to control the number of missed true ALARMS. That is, the number of False Negatives (FN), while keeping the number of False Positives (FP) as low as possible.

The present problem is expressed formally as follows. The protocol of the learning the ALARM rule is spelled out in Alg. (1), while the actions of the *teacher* are spelled out in Table (2). A False Positive (FP) occurs when the learner conjectures an alarm, while careful checking reveals that there is none. A True Positive (TP) happens when the learner predicted ALARM is confirmed by an expert. A False Negative (FN) is the converse, that is the learner thinks that everything works fine in this case, while the situation needs an ALARM in actual fact. A True Negative happens when the learner predicts quite rightly that there is no problem.

	Predicted NOMINAL	Predicted ALARM
NOMINAL	Do Nothing	Adjust rule
ALARM	Do Nothing	HORN

Table 2. Is it possible to learn the prediction rule when only considering the following actions? Especially, when the learner says that everything is normal, no ALARM can be detected. Only when an ALARM is predicted, the operator checks the situation. One might think of this as setting as 'learning from a lazy teacher'.

Assume that $x_t \in \mathbb{R}^d$ for a reasonably small $d > 0$. Let the function $y : \mathbb{R}^d \to \{0, 1\}$ denote wether the teacher would raise ALARM ($y(x_t) = 1$) or NOT ($y(x_t) = 0$). Let the hypothesis of our learner at time t be represented as a vector $w_{t-1} \in \mathbb{R}^d$, such that

$$f_{t-1}(x_t) = I(w_{t-1}^T x_t \geq 0), \quad \forall t, \tag{1}$$

where I is the indicator function where $I(z) = 1$ if z is true, and $I(z) = 0$ otherwise. Now the aim is to find a strategy which results in a sequence $(w_{t-1})_t$ where $w_0 = 0_d$, such that the number of False Negatives (FN) n_- defined as

$$n_- = \sum_{t=1,2,\ldots} y(x_t)\left(1 - f_{t-1}(x_t)\right), \tag{2}$$

is (nearly) zero , while the number n_+ of False Positives (FP) defined as

$$n_+ = \sum_{t=1,2,\ldots} \left(1 - y(x_t)\right) f_{t-1}(x_t), \tag{3}$$

is minimal. Thirdly, all cases of $t : y(x_t) = 1$ occur in the end, that is they are not to be used for learning the rule at first instance. This setting is unlike the traditional classification setting where labels are symmetrical. For completeness, the protocol for classification is given in Alg. (3).

Protocol of learning a classifier:

Given g_0
FOR $t = 1, 2, \ldots$
(1): A measurement x_t comes in.
(2): The learner predicts a label $g_{t-1}(x_t) \in \{-1, +1\}$.
(3): The teacher reveals the true label $z(x_t) \in \{-1, +1\}$.
(4): The learner adjusts $g_{t-1} \to g_t$ if $z(x_t)g_{t-1}(x_t) < 0$.
END

Table 3. The protocol for mistake driven learning of a binary classifier.

The realisation that learning from only one class can be done, comes from interpreting the update rule of the PERCEPTRON, see Fig. (1.b). It is seen that a mistake on case x_t prompts a signed update of the rule as $w_t \pm x_t$. The sign is determined by wether it is a FP or FN. Now, observe that a sign can also be imposed by considering the inverse sample $-x_t$. That is, if a sequence $\{(x_t, y(x_t))\}_t$ works well, then the sequence $\{(y(x_t)x_t, 0)\}_t$ results in the same estimate. However, the latter sequence only requires samples with labels '0' .

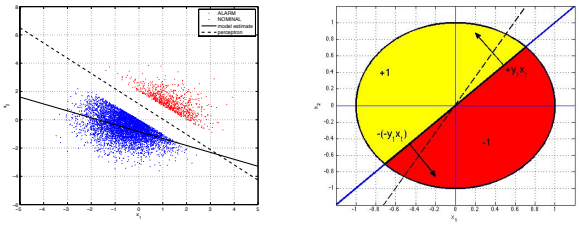

Fig. 1. (a) Example of observations being either ALARM (red) or NOMINAL (blue). The traditional approach when no ALARM samples are available, is to fit a model (solid line) to the NOMINAL data. The PERCEPTRON however focusses only on the margin (dashed line). (b) Visualisation of the PERCEPTRON rule. The dashed straight line is the true rule, the solid line is the learned rule represented as w_t at a certain time instant t. By symmetry of the rule, one can interchange $y_t = +1$ (False Negative) mistakes by $y_t = -1$ mistakes (False Positive), while maintaining exactly the same net outcome.

Now, the desired properties are spelled out. The question becomes which parameter to choose in order to obtain these properties. This question is here answered as follows. While for traditional learning approaches, the margins are taken symmetrical around the decision rule, in this case a proper choice of an asymmetric margin is more plausible. That is, since both classes (ALARM or NOMINAL) have a very different frequency of occurrence, it is reasonable to modify the symmetry of the margin in order to satisfy the constraints on n_+ and n_-. The next subsection works out how this idea can be implemented using a very simple classifier: the PERCEPTRON learning rule.

1.1 AN ASYMMETRIC PERCEPTRON

A first approach to taylor the PERCEPTRON rule to the present context is to tune the asymmetry in order to guarantee properties on the to different types of mistakes. The assumption which makes this analysis possible is the *assumption that such a rule exists.* That is, we have to assume that a perfect ALARM rule exists, and in that case our algorithm will perform with certain guarantees.

PERCEPTRON$_+$
Let $w_0 = 1_d$ and $b_0 = 0$
FOR $t = 1, 2, \ldots$
(1): A measurement $x_t \in \mathbb{R}^d$ comes in.
(2): If $w_{t-1}^T x_t + b_{t-1} > 0$, then issue ALARM. Else, do nothing.
(3): In case of ALARM, query $y(x_t) \in \{0, +1\}$.
(4): If $y(x_t) = 1$ let $w_t = w_{t-1}$ and $b_t = b_{t-1}$.
Else $w_t = w_t + x_t$ and $b_t = b_{t-1} + 1$.
END

Table 4. The protocol for learning a detection rule. The aim of the learner is to control the number of missed true ALARMS. That is, the number of False Negatives (FN), while keeping the number of False Positives (FP) as low as possible.

The main result is a modification of this simple rule which exploits the fact that there are in the beginning only one-sided mistakes possible. The first result is a straightforward application of the Perceptron mistake algorithm. Consider the classification setting where $z(x_t) \in \{-1, 1\}$, but where only one-sided mistakes occur, that is, only values of $z(x_t) = -1$ occur.

Lemma 1. Assume that there exists a $\bar{w}$ such that for all possible x_t one has

$$\begin{cases} (\bar{w}^T x_t) \geq 1 & y(x_t) = 1 \\ (\bar{w}^T x_t) \leq -1 & y(x_t) = 0. \end{cases} \tag{4}$$

Then one has for all $-1 \leq c \leq 1 \in \mathbb{R}$ that

$$n_+(1-c)^2 + n_-(1-c)^2 \leq R^2 \|w\|_2^2 + R^2 c^2. \tag{5}$$

Proof: First, observe that there are many decision rules separating the data points under the above assumptions. For example, every rule given as

$$f(\bar{w}, c) = I(\bar{w}^T x_t + c > 0), \tag{6}$$

for $-1 \leq c \leq 1$ can be used. Note that the new rule is now shifted away from the origin, requiring in general an extra intercept term

$$f(\bar{w}, c) = I(\tilde{w}^T \bar{x}_t > 0), \tag{7}$$

where $\tilde{x}_t = (x_t^T, 1)^T \in \mathbb{R}^{d+1}$ and $\tilde{w} = (\bar{w}^T, c)^T \in \mathbb{R}^{d+1}$. Note that $\|\tilde{w}\|_2^2 = \|\bar{w}\|_2^2 + c^2$. This rule has now margins of size $(1-c)$ in case $y(x_t) = 0$, and $(1+c)$ in case $y(x_t) = 1$. Now the derivation of Block and Novikoff is repeated for arbitrary values of this c. By unfolding the recursion of the learning rule one has

$$w_t = (1-c)^2 \sum_{s \in M_t} x_s, \; b_t = (1-c)^2 \sum_{s \in M_t} 1. \tag{8}$$

Since for any $d \in M_t$, one has $y(x_s) = 0$. The resulting rule is given as

$$f_t(x_t) = I(w_{t-1}^T x_t + b_{t-1} > 0), \tag{9}$$

such that a FP of the rule occurs if $(1 - y(x_t))(w_{t-1}^T x_t + b_{t-1}) > 0$. Note that the value of c is immaterial as long as $c > -1$. Denote $\tilde{w}_t = [w_t, b_t]$, then

$$\tilde{w}_t^T \tilde{w} \leq \|\tilde{w}\|_2 \|\tilde{w}_t\|_2 \leq \|\tilde{w}\|_2 R \sqrt{n_+(1-c)^2 + n_-(1+c)^2}. \tag{10}$$

Conversely, one has

$$\tilde{w}_t^T \tilde{w} \geq n_+(1-c)^2 + n_-(1+c)^2 \tag{11}$$

Combining the inequalities gives the result. Q.O.D.

The resulting optimal rule is hence given by the solution to

$$\min_{c, n_+} n_+ \text{s.t.} \;\; n_+(1+c)^2 + n_-(1-c)^2 \geq R^2 \|\bar{w}\|_2^2 + R^2 c^2, \tag{12}$$

That is, what is the optimal asymmetry allowing for the least number n_+ of FP while ensuring at most n_- FN. This optimisation allows for a closed form solution of c. First, observe that any solution lies obtains equality instead of '<'. Then, lets introduce a Lagrange parameter $\beta \in \mathbb{R}$, rewriting the problem as

$$\min_{c, n_+} \max_{\beta} n_+ + \beta \left(n_+(1+c)^2 + n_-(1-c)^2 - R^2 \|\bar{w}\|_2^2 - R^2 c^2 \right). \tag{13}$$

In case a feasible solution exists (Slater's condition), the problem is equivalent to

$$\max_{\beta} \min_{c, n_+} L(\beta, n_+, c) = n_+ + \beta \left(n_+(1+c)^2 + n_-(1-c)^2 - R^2 \|\bar{w}\|_2^2 - R^2 c^2 \right). \tag{14}$$

This result in the following first order conditions

$$\frac{\partial L(\beta, n_+, c)}{\partial n_+} = 0 \Leftrightarrow \beta = \frac{-1}{(1+c)^2},$$
$$\frac{\partial L(\beta, n_+, c)}{\partial c} = 0 \Leftrightarrow 2c\beta(n_+ + n_- - R^2) + 2\beta(n_+ - n_-) = 0, \tag{15}$$

or

$$c^* = \frac{n_- - n_+}{(n_+ + n_- - R^2)}. \tag{16}$$

Note the fact that $c^* > -1$ for any $n_+ > n_-$. This is quite agreeable since it means that the optimal c^* can be realised in a PERCEPTRON algorithm once the number of FP's is more prevalent than the FN. This is typically true in the present setting. Hence the constraint becomes

$$n_+ \left(\frac{2n_- - R^2}{n_+ + n_- - R^2} \right)^2 + n_- \left(\frac{2n_+ - R^2}{n_+ + n_- - R^2} \right)^2 - R^2 \|\bar{w}\|_2^2 - R^2 c^2 > 0. \tag{17}$$

From these derivations, the following result follows

Lemma 2. If $n+ \geq n_- > \frac{R^2}{2}$, then

$$-1 < c^* < 1. \tag{18}$$

And hence the optimal c is realisable in the PERCEPTRON rule.

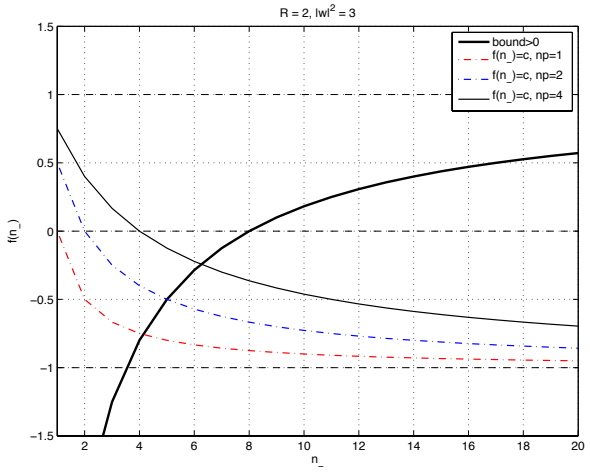

Fig. 2. Evolution of the optimal c and the bound for various values of n_+.

1.2 A FILTRON RULE

The above ALARM rule is based on finite dimensional vectors x_t. However, many engineering settings suggests the use of filters to compute such a rule. Then the question becomes how to learn such a filter from observations. Ideas of the symmetrical PERCEPTRON are moved towards this new context, that is, this subsection treats only the simple case where margins are equal at both sides. This subsection will spell out some of the ideas, provide evidence for the result, and motivate later integration with the above asymmetrical rule.

At first, the form of the general solution is given. Suppose that one is monitoring a system based on a signal $\{u_t\}_t$. The reference filter is given as

$$y_t = I\left(\sum_{\tau=0}^{\infty} \bar{h}_\tau u_{t-\tau} > 0 \right). \tag{19}$$

We will denote the infinite vector $\bar{h} = (\bar{h}_0, \bar{h}_1, \bar{h}_2, \dots)$, so that one can write equivalently $y_t = I(\bar{h} * u_t > 0)$. Straightforward application of the above derivations would yield a guarantee which deteriorates linearly in d (the length of the filter). This is clearly a bad idea when $d \to \infty$. That is, when up sampling twice, the norm of the impulse response increases approximately twice as well.

However, there exists a nice way around this using basic results in systems and realisation theory, see e.g. Kailath [1980]. This is based on the (symmetrical) Hankel matrix, defined as

$$H_d(h) = \begin{bmatrix} h_0 & h_1 & h_2 & \dots & h_{d-1} \\ h_1 & h_2 & h_3 & & \\ h_2 & h_3 & h_4 & & \\ \vdots & & & \ddots & \\ h_{d-1} & & & \dots & h_{2d-1} \end{bmatrix}. \tag{20}$$

Given an impulse response vector h, the rank of the corresponding H_d for arbitrary large value of d equals the McMillan degree of the minimal realisation, see e.g. Fazel et al. [2001], Recht et al. [2010], Liu and Vandenberghe [2009] and references. Secondly, we spell out how the convolution of h with a signal u_t can be rephrased using this matrix. Let z_t be defined as follows

$$z_t = \sum_{\tau=0}^{n} \bar{h}_\tau u_{t-\tau} = \text{trace}\left(H_d(\bar{h}) U_d(t) \right), \tag{21}$$

where $U_n(t)$ is defined as the symmetrical matrix

$$U_d(t) = \begin{bmatrix} u_t & \frac{u_{t-1}}{2} & \frac{u_{t-2}}{3} & \dots & \frac{u_{t-d+1}}{d} \\ \frac{u_{t-1}}{2} & \frac{u_{t-2}}{3} & \frac{u_{t-3}}{4} & & 0 \\ \frac{u_{t-2}}{3} & \frac{u_{t-3}}{4} & \frac{u_{t-4}}{5} & & 0 \\ \vdots & & & \ddots & \\ \frac{u_{t-d+1}}{d} & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{22}$$

Some elementary matrix algebra shows that

$$\|U_n(f)\|_F^2 = \sum_{\tau=0}^{d} \frac{\tau u_{t-\tau}^2}{\tau^2} \leq R^2 \ln(d+1). \tag{23}$$

where one has $u_t^2 \leq R^2$ for any t. Note that this assumes that the impulse response h is (or can be approximated) of finite length $d < \infty$. This is not too restrictive, as the prediction has some margin (allowing approximations) anyway.

The FILTRON rule is spelled out in table (5). This algorithm comes with the following guarantee:

Lemma 3. Assume that there exists a $\bar{h}$ such that one has for any t that

$$\begin{cases} \text{trace}(H_d(\bar{h})^T U_d(t)) \leq -1 & y_t = 0 \\ \text{trace}(H_d(\bar{h})^T U_d(t)) \geq 1 & y_t = 1, \end{cases} \tag{24}$$

and such that $H_d(\bar{h})$ is of rank bounded by r.

$$n_+ + n_- \leq \|H_d(\bar{h})\|_*^2 R^2 \ln(1+d) \tag{25}$$

Proof: The evolution of

$$\text{trace}(H_d(\bar{h})^T H_d(h_t)), \tag{26}$$

FILTRON

Let $w_0 = 1_d$ and $b_0 = 0$
FOR $t = 1, 2, \ldots$
(1): A new signal $u_t \in \mathbb{R}^d$ comes in.
(2): If $\mathcal{P}_*(h_{t-1}) * u_t > 0$, then issue ALARM. Else, do nothing.
(3): In case of ALARM, query $y_t \in \{0, +1\}$.
(4): If $y_t = 0$, raise a FP and update the rule as
$h_t = h_{t-1} + (u_t, \ldots, u_{t-d+1})^T$.
Else $h_t = h_{t-1}$.
END

Table 5. the FILTRON algorithm. Note that we include $\mathcal{P}_*(h_{t-1})$ for predicting wether a new case is NOMINAL or deserves an alarm. This is the PROXIMAL mapping for the Nuclear norm. The resulting scheme is *not* a projected gradient descent scheme, as the result of this projection is only used to make predictions, and is not used in the recursion itself.

is bounded from both sides. Note that by unfolding the recursion, one has

$$H_d(h_t) = \sum_{s \in M_{t-1}} z_t U_d(s), \tag{27}$$

where we defined $z_t = 2y_t - 1$ for all t. So that

$$\begin{aligned} \operatorname{trace}(H_d(\bar{h})^T H_d(h_t)) &= \operatorname{trace}\left(\sum_{s \in M_{t-1}} z_t H_d(\bar{h})^T U_d(s) \right) \\ &= z_t \sum_{s \in M_{t-1}} \operatorname{trace}\left(H_d(\bar{h})^T U_d(s) \right) \geq n_+ + n_-. \end{aligned} \tag{28}$$

The upper-bound follows from the following result on the trace result (the cyclic property and the inequality see e.g. Coope [1994]), Let $A, B \in \mathbb{R}^{d \times d}$ be squared matrices, then

$$\operatorname{trace} AABB \leq \operatorname{trace} AA \operatorname{trace} BB. \tag{29}$$

Let $A = H_d(\bar{h})$ and $B = H_d(h_t)$, then

$$\operatorname{trace}(H_d(\bar{h})^T H_d(h_t))^2 \leq \|H_d(\bar{h})\|_F^2 \|H_d(h_t)\|_F^2. \tag{30}$$

Now we use the inequality that for any matrix A of rank r, one has $\|A\|_F \leq \|A\|_* \leq \sqrt{r}\|A\|_F$. Hence

$$n_+ + n_- \leq \|H_d(\bar{h})\|_* \|H_d(h_t)\|_F. \tag{31}$$

Working out the last term using eq. (27), the definition of a mistake and eq. (23) gives

$$\|H_d(h_t)\|_F^2 \leq (n_+ + n_-) R^2 \ln(d+1). \tag{32}$$

Combining this inequality with eq. (30) yields the result. Q.O.D.

The same reasonings as in the previous section are directly applied to handle asymmetrical margins, or learn only from FPs.

Note that this bound is qualitatively different from what straightforward application of the PERCEPTRON derivation would give us. The main difference is in the norm $\|U_d(t)\|_F^2$ which results in a $\ln d$ factor. In case the vector PERCEPTRON rule were use, the term $\|(u_t, \ldots, u_{t-d+1})\|_2^2$ which grows linear in d would be unavoidable. In order to indicate that this difference is not compensated by use of the Nuclear norm of the Hankel matrix as compared to the 2-norm $\|h_0\|_2^2$, the norms of different examples are given in the experimental section. A proper theoretical investigation of the order of the nuclear norm of the Hankel matrix of small rank will be presented later.

2. NUMERICAL EXPERIMENTS

Two numerical examples are presented in order to illustrate the results. They are based on artificial setups, and need further tuning in more realistic cases. Nevertheless, they are included as they give some insights in the methods.

2.1 LMS versus PERCEPTRON$_+$

Firstly, we illustrate the difference between a classical approach based on LMS, and that one of the PERCEPTRON. The following experiment was conducted. $n = 5000$ data points $x_t \in \mathbb{R}^d$ are generated from a standard distribution with identity variance and zero mean, and they are assigned to NOMINAL or ALARM by use of a rule of a linear rule $1_d^T x_t \geq 1$. It is made sure that the decision rule has a margin of size $\rho > 0$. The results for various dimensions d and margins ρ are given in Table 6.

(d, ρ)	LMS	RLS	PERCEPTRON$_+$
$(2, 1)$	(0.1555, 0.0200)	(0.0594, 0.0001)	(0.0005, 0.0068)
$(2, 0.1)$	(0.2312, 0.2305)	(0.1037, 0.1572)	(0.0023, 0.0105)
$(5, 1)$	(0.3473, 0.2909)	(0.1148, 0.0793)	(0.0014, 0.0067)
$(5, 0.1)$	(0.3472, 0.3744)	(0.1305, 0.3412)	(0.0166, 0.0396)
$(10, 1)$	(0.6055, 0.2626)	(0.1372, 0.2954)	(0.0050, 0.0128)
$(10, 0.1)$	(0.5870, 0.2823)	(0.1405, 0.4769)	(0.0298, 0.0545)
$(50, 1)$	(0.9837, 0.0160)	(0.1513, 0.6198)	(0.0386, 0.0540)
$(50, 0.1)$	(0.9846, 0.0147)	(0.1535, 0.6676)	(0.0716, 0.0954)

Table 6. Numerical performances observed in the first experiment, relating a traditional approach of adapting to the data in the NOMINAL case, versus the performances obtained by the PERCEPTRON$_+$ case. The modelling approach can be done in many different ways, here we implement an adaptive LMS filter and an RLS approach. Given are the fractions $FN/(TP+FN)$ and $FP/(TN+FP)$ recorded during the adaption/learning process. It is seen immediately that the proposed rule deals orders of magnitudes better with this situation, especially for increasing d and for the portion of FN (missed ALARMS). Note that a LMS approach of comparable complexity gives very poor results.

2.2 PERCEPTRON versus the FILTRON

Secondly, numerical evidence is provided for the task of inferring a filter with the FILTRON as opposed to a straightforward application of the PERCEPTRON to this case. This experiment constructs a signal u_t as white noise of unit variance and zero mean. The ultimate rule is given as

$$y_t = I\left(\bar{h} * u_t \geq 0\right), \; h(q) = \frac{1 + 0.9q^{-1}}{1 - 0.9q^{-1}}. \tag{33}$$

A typical difference of the result is given in Fig. (4). The norms of the *true* filter $\bar{h}$ are given as follows. The 2-norm $\|\bar{h}\|_2^2 = 1.6510$, the Frobenius norm of the associated Hankel matrix $\mathcal{H}(\bar{h})$ is $\|\mathcal{H}(\bar{h})\|_F^2 = 3.0477$, while

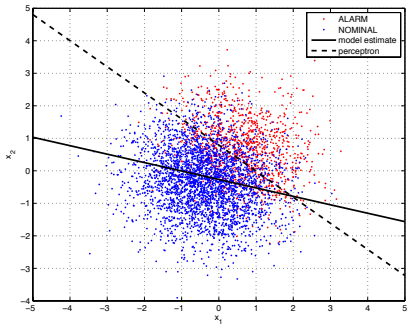

Fig. 3. Typical observed case in the first experiment, here for $d = 5$ and $\rho = 1$. While traditional methods try to find regularities (i.e. a model) in the NOMINAL data, the PERCEPTRON only focusses on the separating hyperplane.

the Nuclear norm is $\|\mathcal{H}(\bar{h})\|_*^2 = 3.0477$ as well. Note that neither scales up too bad in terms of d (here $d = 25$). On the average, the performances of the PERCEPTRON rule and the FILTRON rule go as follows. The PERCEPTRON achieves a portion of $FP/(FP + TN) = 0.0243$ and $FN/(FN + TP) = 0.0199$. The FILTRON achieves $FP/(FP + TN) = 0.0159$ and $FN/(FN + TP) = 0.0133$. This number is achieved after seeing $n = 5000$ samples.

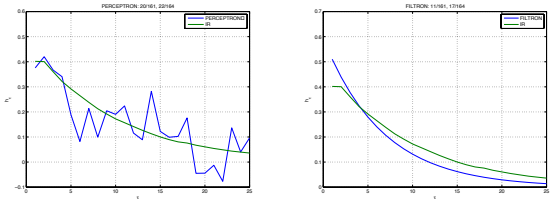

Fig. 4. (a) Resulting estimate when applying the standard PERCEPTRON rule. (b) Resulting estimate when applying the FILTRON rule. One sees by comparison with the previous picture that the estimate is much smoother, resulting from use of the nuclear norm and its relation to the rank of the Hankel matrix associated to the filter.

3. CONCLUDING REMARKS

This paper investigates the question wether one can learn an ALARM rule without having encountered a FAULT as yet. It turns out that one can by avoiding making stochastic assumptions. Doing so would imply that we can merely state results about *typical* situations, while a FAULT is by definition not. However, by resorting to the modern theory of online learning and mistake-driven learning one can get around this point. Furthermore, two variations of the PERCEPTRON rule are introduced. Firstly, the standard rule is adapted to the asymmetrical case in order to provide guarantees on the number of FN and FP as desired. Secondly, the rule is extended to the filter case by making use of the Nuclear norm heuristic of the Hankel matrix associated to the filter estimate.

There are ample of questions left for open at this point. The main theoretical issue is why the PROX mapping $\mathcal{P}_*(h_{t-1})$ in the FILTRON algorithm of Table 5 is reducing the number of mistakes, as empirical evidence seems to suggest. Answers are by now appearing for use of the Nuclear norm in the batch case (see e.g. Recht et al. [2010]), but it is as such completely unknown what happens in online cases (see e.g. Cesa-Bianchi and Lugosi [2006] for a survey of results). It is also well-known that the PERCEPTRON rule can handle the non-departing case as well - see e.g. Mohri et al. [2012]. Extension of this work to the present setting provides a new challenge. Finally, it is clear that deeper insight in the methods will be gained from more extensive simulation and study of case studies.

REFERENCES

Michèle Basseville. Detecting changes in signals and systemsa survey. *Automatica*, 24(3):309–326, 1988.

N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

I.D. Coope. On matrix trace inequalities and related topics for products of hermitian matrices. *Journal of mathematical analysis and applications*, 188(3):999–1001, 1994.

Maryam Fazel, Haitham Hindi, and Stephen P Boyd. A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference, 2001. Proceedings of the 2001*, volume 6, pages 4734–4739. IEEE, 2001.

Janos Gertler. *Fault detection and diagnosis in engineering systems*. CRC press, 1998.

B. Hassibi, A.H. Sayed, and T. Kailath. H_∞ optimality of the LMS algorithm. *IEEE Transactions on Signal Processing*, 44(2):267–280, 1996.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, Heidelberg, 2001.

Thomas Kailath. *Linear systems*, volume 1. Prentice-Hall Englewood Cliffs, NJ, 1980.

Steven M Kay. *Fundamentals of Statistical signal processing, Volume 2: Detection theory*. Prentice Hall PTR, 1998.

Erich Lehmann. *Nonparametrics: statistical methods based on ranks (POD)*. Prentice-Hall: 1st edition (1975). Springer (Berlin): Revised edition, 2006.

Erich L Lehmann. The fisher, neyman-pearson theories of testing hypotheses: One theory or two? *Journal of the American Statistical Association*, 88(424):1242–1249, 1993.

Z. Liu and L. Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1235, 2009.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. The MIT Press, 2012.

B. Recht, M. Fazel, and P.A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

Harry L Van Trees. *Detection, estimation, and modulation theory*. Wiley. com, 2004.

Mathukumalli Vidyasagar. *A theory of learning and generalization*. Springer-Verlag New York, Inc., 2002.