From elementary martingale calculus to rigorous properties of mixtures of experts

Badih Ghattas & Gonzalo Perera

Abstract

Authors have performed learning algorithms, based on mixtures of experts, who achieve a good performance under severe time/cost restrictions, and that can be applied to non-stationary data. This is of particular interest for applications like quality of Service (QoS) prediction on IP data networks (see [12]). In this paper we show how can all the properties of this algorithms be proved in a strictly rigorous manner, with no other tools that elementary martingale theory at hand.

Mathematics Subject Classification (2000): 68T05,93E35, 62J20.
Key words: learning algorithm, prediction model, classification, regression, martingales, mixture of experts, limit theorems.

1 Introduction: motivation and basic ideas on Supervised Learning Algorithms.

1.1 On-line prediction for non-stationary engineering data.

In this paper we show how very simple probabilistic tools give rigorous proof for learning algorithms developed to solve real Engineering Applications. It is quite usual in real Engineering problems to use machine learning algorithms, or more precisely, supervised learning algorithms (see details in next subsection) to control a process. Of course, those algorithms must be as efficient and low-cost as possible. But many real engineering sets exhibit a clear non-stationary behavior, which, a priori, are out of the scope of the most usual learning algorithms.

For an example, on-line estimation of Data Networks performances is crucial to guarantee Quality of Service (QoS) for multi-purpose networks, whose traffic is usually non-stationary at all the possible time scales (see [14], [20]).

Algorithms for risk analysis of credit cards operations (see [5]) or surveillance of atmospheric pollution (see [3]) are other examples of Engineering problems where the same type of requirements (efficiency at the same time than low cost

and short computation time for non-stationary data) appears.

Coming back to the example of QoS prediction in Data Networks, large deviation principles (based in the notion of *effective bandwidth* as presented in [15]) have produced a series of results allowing to predict QoS at a given link, at the level of the backbone of the network (see for instance [1], [16]) and some partial results allowing to assure QoS from end-to-end (see for instance [4],[6]).

A relevant current of research in Data Networks has been active measurement of end-to-end QoS via probe packets. The basic idea is that if one sends some packets, the observed delay at their arrival will allow to estimate how heavy is the current traffic in the Network and what level of QoS can be assured. Even if the methodology of probe packets is not universally appliable (see [7]), for some particular Networks and parameters, this idea has shown to be successful. In [2] a functional regression method for non-stationary and dependent data was developed, providing a Learning Machine algorithm that, given the empirical distribution of the delay of the probe packets, predicts the QoS for a video or any other heavy network process that one wants to run. In [12], a much more general learning strategy for non-stationary data has been built up, based on mixtures of experts with different skills.

However, there was not a detailed, rigurous proof of the results provided in [12]. In this paper, we present a careful proof of the properties of this method. We will prove the results in the simplest possible context. In this way, an elementary knowledge of martingale calculus and its limit theorems will be largely enough to understand most of this paper. The extension from this context to the general setting of [12] is an exercise for the reader that knows Machine Learning Theory well, and is a merely technical effort for the general reader. We hope that our choice will make this work readable for a wide mathematical public that, hopefully, may feel interested by Machine Learning or Data Network Performance matters.

To get started in the following section we present the general framekork for *Supervised Learning Algorithms*.

1.2 Basic ideas of Supervised Learning.

Let P denote a probability on an underlying probability space $(\Omega, \mathcal{A})$, where the couple (X, Y) is defined, where X takes values in an arbitrary mesurable space S_X and Y takes values on a measurable space S_Y.

Some notation we will use:

- ρ is the joint distribution of the couple (X, Y), that is, for any measurable sets $A \subset S_X$ and $B \subset S_Y$, we have

$$\rho(A \times B) = P(X \in A, Y \in B).$$

- $p(./X)$ denotes the conditional probability distribution of Y given X, defined in a rigurous way as the almost surely (a.s., for short) unique measurable function of X satisfying:

$$E(1_{\{Y \in B\}} 1_{\{X \in A\}}) = E(p(B/X) 1_{\{X \in A\}})$$

  for any measurable sets A and B. We assume that this conditional distribution is regular, what is true if S_X, S_Y are standard spaces.

- π is the marginal law of X (i.e., $\pi(A) = P(X \in A)$); then we can write

$$\rho(A \times B) = \int_A p(B/x)\pi(dx) = \int_A \int_B p(dy/x)\pi(dx).$$

We assume that both ρ and p are unknown. In this paper we also assume that π is unknown but the key point of the prediction problem concerns the process of *learning* p (and hence, ρ).

In the prediction problem, we observe the value of $X(\omega)$ and we are requested to guess the value of $Y(\omega)$. We will often call X the *input* or the *pattern*, and Y *output* or *label*. In general, our prediction will be $f(X(\omega))$ where f is a measurable function from S_X on S_Y, that we call *predictor*. The main problem is to find a "good"predictor, what previously requires to set a criterion to determine whether a given predictor is "good"or not.

If we have a criterion to quantify how much we "loose"by predicting a value u for $X(\omega) = x$ where the true value was $Y(\omega) = y$ and this quantification is denoted by $L(x, u, y)$, we may introduce the *loss function*

$$L : S_X \times S_Y \times S_Y \Longrightarrow \mathbb{R}.$$

We will assume in the sequel that $L(x, u, y) \geq 0$ and that $L(x, u, y) = 0$ if and only if $u = y$.

From now on (X, Y) denotes a generic random vector distributed according to ρ. As said before, a *predictor* or *prediction rule* is in general a measurable function $f : S_X \Longrightarrow S_Y$ and its quality is measured by means of the expected loss:

$$\tau_L(f) = E\{L(X, f(X), Y)\} = \int_{S_X} \int_{S_Y} L(x, f(x), y)\rho(dx, dy) =$$

$$\int_{S_X} \{ \int_{S_Y} (L(x, f(x), y)) p(dy/x) \} \pi(dx).$$

Hence, we say that a predictor f is better than a predictor g if $\tau_L(f) \leq \tau_L(g)$. For instance, if we take $L(x, u, y) = 1_{\{u \neq y\}}$ then $\tau_L(f) = P(f(X) \neq Y)$ (the *overall error rate*).

As another classical example, if S_Y is a normed space, $\|.\|$ denotes its norm, we assume that $E\{\|Y\|^2\} < \infty$ and set $L(x, u, y) = \|u - y\|^2$, then $\tau_L(f) = E\{\|f(X) - Y\|^2\}$ (the *mean integrated squared error*, MISE, for short).

In general, if we assume that, for any x in a set of π-probability one, there exists an unique value $f^*(x)$ such that

$$\int_{S_Y} L(x, f^*(x), y) p(dy/x) \leq \int_{S_Y} L(x, u, y) p(dy/x) \text{ a.s. with respect to } u \in S_Y,$$

and if the function $f^* : S_X \Longrightarrow S_Y$ is measurable, then f^* is the optimal predictor, since a straightforward computation shows that $\tau_L(f^*) \leq \tau_L(f)$ for any predictor f.

In the case of the overall error rate and when $p(./x)$ is unimodal, $f^*(x)$ is called *conditional mode* or *Maximum A Posteriori* (MAP) given x, and corresponds to the value of u that maximizes $p(u/x)$. In the case of the MISE, $f^*(X) = E(Y/X)$. If, for instance, S_Y is a topological vector space and L satisfies some regularity and convexity conditions with respect to u, the existence of $f^*(x)$ can be shown.

The problem is that, in general, one is not able to look for a predictor on the whole set of functions from S_X to S_Y (that may be a very huge set) but only on a given class of functions $\mathcal{F}$ that corresponds to the kind of predictors that we may practically compute. In such a case, the optimal predictor f^* may be not included in $\mathcal{F}$ and , therefore, the best predictor that we will be able to find is f^{**}, such that

$$f^{**} = \text{argmin}_{f \in \mathcal{F}} \tau_L(f).$$

As we will see later, this predictor f^{**} is not available in practice, since the law ρ is unknown and should be estimated from data. Thus, one is not able to minimize τ_L but only an empirical estimation of it.

Remark 1.1: It is also a common practice (and in fact, this will be our case), to consider a *reward function* $R(x, u, y)$, giving the reward to be assigned to a prediction of u for the value x when the real value is y, instead of the loss function. Despite of the fact that the practical motivation may make one approach more appealing than the other, from the mathematical point of view, they are

completely equivalent, since if L is a loss function and C is a suitable constant, then $C - L$ is a reward function.

Remark 1.2: It is also common (and wise), in practice, to make use of the advice of experts, that may be human experts or previously tested algorithms. In our case we make use of experts advice, and we think an expert as a transition matrix $A(./.)$, that gives, for any input x, the probability $A(y/x)$ that this expert assigns to the output y. If an expert is used just by means of a MAP procedure, then, we consider that his answer for an input value x is the (unique) value $a(x) \in S_Y$ such that $A(a(x)/x) \geq A(y/x)$ for any y (if such an $a(x)$ is not unique, then some ordering or sampling procedure may be used to choose only one).

In *supervised learning*, the predictor $\hat{f}_n$ that we can use in practice, is based on a *training sample* $(X_1, Y_1), ..., (X_n, Y_n)$, often assumed to be independent and identically distributed (*iid*, for the sequel) according to the law ρ. If a class $\mathcal{F}$ of functions is used, then we take as our prediction rule $\hat{f}_n$, the element of $\mathcal{F}$ that minimizes

$$\tau_{L,n}(f) = \frac{1}{n} \sum_{i=1}^{n} L(X_i, f(X_i), Y_i),$$

i.e.

$$\hat{f}_n = \text{argmin}_{f \in \mathcal{F}} \tau_{L,n}(f).$$

As an estimation of the performance of this prediction rule, we might take $\tau_{L,n}(\hat{f}_n)$, but this is usually a biased estimation: it overestimates the performance of the predictor. This is also related to what is usually called "overfitting": if, for instance, $\mathcal{F}$ is as big as the whole family of functions from S_X to S_Y, and $X_1, ..., X_n$ contains n different values, it is clear that $\hat{f}_n(X_i) = Y_i$ for any i and that $\tau_{L,n}(\hat{f}_n) = 0$ (perfect fitting over the trainig sample), but when $\hat{f}_n$ is applied to new data the result may be catastrophic (the prediction follows so closely the particular features of the training sample, that it is statistically very poor). This type of problems is detected if the performance of the predictor rule is measured by means of a new sample, called the *evaluation sample*, which is another *iid* sample of the distribution ρ, $(X_1^v, Y_1^v), ..., (X_m^v, Y_m^v)$, independent with respect to the training sample, and the performance of our predictor is estimated by means of

$$\tau_{L,m}^V(f) = \frac{1}{m} \sum_{i=1}^{m} L(X_i^v, f(X_i^v), Y_i^v).$$

Another type of performance estimation, based on well-known procedures such as cross-validation, bootstrap and other resampling techniques, may be used

in practice to give unbiased and numerically efficient estimations of the performance, but we refer to [13] for an extensive account.

Finally, it should be noticed that when the best of all predictors is f^* and the best of possible predictors on our class is f^{**}, the real predictor we use in practice is $\hat{f}_n$. The loss of perfomance due to the difference between f^* and f^{**} is of modellistic nature, it depends on how clever is our choice of $\mathcal{F}$. If a bad choice of $\mathcal{F}$ is made, no further sampling allows to overcome this loss of performance. This is why the difference $f^* - f^{**}$ is often called *approximation error*. On the other hand, the second loss of performance, due to the difference between f^{**} and $\hat{f}_n$ is purely of statistical nature. If very large training samples were available (i.e., if n tends to infinity), under suitable hypothesis on the model (see for instance [9], [10], [18] for a general exposition), $\hat{f}_n$ tends to f^{**}. This explains why the difference $f^{**} - \hat{f}_n$ is often called *estimation error*.

At last, but not least, one must mention the fact that Machine Learning, and, in particular, Supervised Learning Techniques, are also used as means to gains insights on how does intelligence works. That is why the same subject is also presented under the more appealing title of "Artificial Intelligence". In fact, Steven Smale, when requested to list the 18 most relevant mathematical problems for the XXI century, included as Problem 18 the limits of the intelligence, and if it was possible to model and describe how does intelligence evolves (see [17]).

2 General description of the algorithm.

We will now describe the particular characteristics of our learning algorithm. As we said in the introduction, we will develop the whole method in the simplest case. Therefore, we will assume from now on that X takes values in a finite set $S_X = \{1, ..., I\}$ and that Y takes values in $S_Y = \{1, .., J\}$.

Despite the huge variety of procedures that have been proposed for supervised learning (linear methods, neural networks, CART, SVM, Boosting) most of them do work well under the condition that the size of the training sample (n), is assumed to be arbitrary large. This means that massive information is available, allowing to drastic reduction of the estimation error and, with suitable modeling, very efficient learning (see, for instance, [8],[10], [13]). This is clearly not possible for on-line applications or for applications that must exhibit a minimum delay to give a response. And that is the case of our motivating example of QoS prediction for Data Networks. We overcome this difficulty by means of an iterative procedure, that uses a limited ammount of information at each step, and that makes use of a mixture of experts.

Indeed, as predictors, we will have at hand k experts $A_1, ..., A_k$ and a class of

models $\mathcal{F}$. The experts will be fixed and will not change their behavior trough the whole process: given one expert A_i and an input x at any step of the algorithm, the expert always give the same advice and predict the same value of y. We call *advisors* both experts and the optimal predictor chosen from $\mathcal{F}$. Hence, we have $k+1$ advisors, where indexes $1, ..., k$ correspond to the experts and the index 0 to the model. It must be noticed that while $A_1, ..., A_k$ do not change over the whole execution of our algorithm, the specific function f_j selected in $\mathcal{F}$ at step j, changes from one step of the algorithm to the following. As explained in [12], the fact that the optimal predictor of the model is "fresh"(chosen again) at each step of the algorithm, helps to achieve better performances when dealing with non-stationary data and its a major difference with standard sequential procedures. We assume that a familly of $k+1$ reward functions is given. More precisely, denote $H = \{0, ..., k\}$; we consider a function

$$R : S_X \times S_Y \times S_Y \times H \Longrightarrow \mathbb{R}$$

such that for any $h = 0, ..., k$, $R(., ., ., h)$ is a reward function with $R(x, u, y, h)0$ if $u = y$ and $R(x, u, y, h) < 0$ if $u \neq y$. $R(x, u, y, h)$ represents the reward to be asigned to the advisor h if he assigns for $X = x$ the value u when the true value was $Y = y$.

A central role in our algorithm is played by the *credit matrix*.

$$(c_j(x, h))_{x \in S_X, h \in H}$$

which encodes for the step j our confidence in the advisor h to predict the output of x. More precisely denote:

$$h_j(x) = \text{argmax}_{h \in H} c_j(x, h),$$

the most credible advisor to predict x at step j (if there is more than one value of h where the maximum is reached, we may choose for instance the biggest of such values). Then :

- At step j, the prediction of the value to assign to x is done by the advisor $h_j(x)$ (recalling that the case $h_j(x) = 0$ corresponds to the model, i.e., $\hat{f}_j(x)$.

Once the training sample to be used at the step j is available, $(X_1^j, Y_1^j), \ldots, (X_T^j, Y_T^j)$ (which is assumed to be *iid*, following the law ρ), we choose $\hat{f}_j$ as the best candidate in $\mathcal{F}$ according to the following criteria:

$$\hat{f}_j = \text{argmax}_{f \in \mathcal{F}} \Gamma_T^j(f)$$

where,

$$\Gamma_T^j(f) = \frac{1}{T} \sum_{i=1}^{T} R(X_i^j, f(X_i^j), Y_i^j, 0)$$

is the empirical version of the *expected reward*

$$\Gamma(f) = E\{R(X, f(X), Y, 0)\} = \sum_{x \in S_X, y \in S_Y} R(x, f(x), y, 0)\rho(x, y)$$

(Γ is analogous to the the expected loss τ_L if we think in terms of a loss function L instead of a reward function).

Remark 2.1: Even if our method has been inspired from learning problems on network administration, where a merely objective learning seems to be adequate, it seems to be appealing to apply this system in a subjective context, for instance, for behavioral systems (see [11]). To allow the expression of subjective profiles, we need that different advisors gain different reward by a given decision (further, this difference on the credited rewards may be taken as patterns to identify such profiles). That is why we have included the "h "component on the function R and that is why we prefer to speak about "reward"instead of "loss "or "cost".

Once the model has been fitted, we proceed to the validation of the prediction rule and we update the credit matrix. Since each expert $A_1, ..., A_k$ uses a MAP criterion, we denote by $y_1(x), ..., y_k(x)$ the answer that each expert gives to the input x. In the cycle j of our algorithm, we use a validation sequence $(X_1^{v,j}Y_1^{v,j}), ..., (X_V^{v,j}Y_V^{v,j})$ (*iid* and distributed according to ρ), independent with respect to the training sequence of the same cycle j and independent with respect to both training and validation samples of previous cycles.

Then the credits are updated as following:

- For each $i = 1, ..., V$, compute $h_j(X_i^{v,j})$.
- Compute the prediction for each observation of the validation sequence by means of $y_h(X_i^{v,j})$ if $h_j(X_i^{v,j}) = h \geq 1$ or by means of $\hat{f}_j(X_i^{v,j})$ if $h_j(X_i^{v,j}) = 0$. In any case, let us denote by $f_j(X_i^{v,j})$ the predicted output.
- Update the credits as follows

$$c_{j+1}(x, h) = c_j(x, h) + \frac{1}{V} \sum_{i=1}^{V} R(X_i^{v,j}, f_j(X_i^{v,j}), Y_i^{v,j}, h) 1_{\{h_j(X_i^{v,j})=h, X_i^{v,j}=x\}}$$

Remark 2.2: Observe that to update the credit $c_j(x, h)$ we only use the observations of the validation sample where the input was x and the most credible expert was h. In particular, if a value of x does not appear in the validation sample, its credit is not changed, and if a given expert was less credible than

others for any input of the validation , its credit does not change.

Some final remarks on general notation: we use the symbol ":="for a definition that is set inside an equation. If $\mathcal{C}$ is a collection of random variables, $\sigma(\mathcal{C})$ denotes the σ-algebra generated by $\mathcal{C}$. If $\mathcal{F}, \mathcal{L}$ are σ-algebras on Ω, $\mathcal{F} \vee \mathcal{L} := \sigma(\mathcal{F} \cup \mathcal{L})$. As usual, convergence in law may be thought both at the level of random variables or at the level of probability distributions and notation may mix both levels. For instance, if $Z_1, ..., Z_n, ..$ is a sequence of random variables and we state

$$\lim_n Z_n = N(0,1) \text{ in law },$$

we are saying that, with respect to the topology of the weak convergence of probability measures, the sequence of distribution measures P^{Z_n} converges to a standard gaussian probability measure.

3 Theoretical results.

In this section we derive the asymptotic behavior of our Restricted Resources Learning Algorithm (RRLA, for short), when the number of iterations tends to infinity.

We will first obtain the limit of the credit matrix $(c_j(x,h))_{x \in S_X, h \in H}$ when j tends to infinity. Then we will compare the performance of the RRLA to that of an algorithm based on the whole set of training sequences (i.e., with No Restriction: we will call this algorithm NRLA, for short). In particular we will show that, under reasonable assumptions, RRLA behaves almost as well as NRLA, but with very lower cost and computation requirements, and therefore it can be seen as a performant alternative that respects restrictions.

Let us set some notation and assumptions.
First of all, to avoid trivialities, we assume that $\pi(x)0$ for any $x \in S_X$. For each one of the experts indexed by $h = 1, ..., k$ and any $x \in S_X, y \in S_Y$, we define

$$r_h(x,y) = R(x, y_h(x), y, h)$$

$$r_h(x) = E\{r_h(X,Y)/X = x\} = \sum_{y \in S_Y} r_h(x,y) p(y/x)$$

With the notation of the end of the previous section, denote

$$f^{**} = argmax_{f \in \mathcal{F}} \Gamma(f), \hat{f}_j = argmax_{f \in \mathcal{F}} \Gamma_T^j(f).$$

(We assume again that those maximum values are attained at a unique element of $\mathcal{F}$).
Define, for any $x \in S_X, y \in S_Y$:

$$r_0(x,y) = R(x, f^{**}(x), y, 0)$$

$$r_0(x) = E\{r_0(X,Y)/X = x\}$$
$$r_0^j(x,y) = R(x, \hat{f}_j(x), y, 0)$$
$$r_0^j(x) = \sum_{y \in S_Y} r_0^j(x,y) p(y/x)$$

Observe that $r_0^j(x,y)$ (resp. $r_0^j(x)$) is a random function of (x,y) (resp. x). Let us also call $\mathcal{S}$ the set of all the functions from S_X to S_Y. If (X,Y) is a random vector independent of the training sample and distributed according to ρ, we have that:

$$\begin{aligned} E\left(r_0^j(X)\right) &= E\left(E\{R(X, \hat{f}_j(X), Y, 0)/\hat{f}_j\}\right) \\ &= \sum_{f \in \mathcal{S}} E\{R(X, \hat{f}_j(X), Y, 0)/\hat{f}_j = f\} P(\hat{f}_j = f) \\ &= \sum_{f \in \mathcal{S}} E\{R(X, f(X), Y, 0)\} P(\hat{f}_j = f) \\ &= \sum_{f \in \mathcal{F}} \Gamma(f) P(\hat{f}_j = f) \qquad (1) \end{aligned}$$

In the last equality, we have used the fact that $\hat{f}_j \in \mathcal{F}$. In addition, we clearly have that:

$$E(r_0^j(x)) = \sum_{f \in \mathcal{F}} \sum_{y \in S_Y} R(x, f(x), y, 0) p(y/x) P(\hat{f}_j = f).$$

We assume from now on the following hypothesis:

- (H1) For any $x \in S_X$, there exists an unique $h(x) \in H$, such that

$$r_{h(x)}(x)0, r_h(x) < 0 \text{ if } h \neq h(x).$$

Remark 3.1: Observe that if for a given value x_0, $r_h(x_0)$ depends only on the values of $R(x_0, ., ., h)$. Hence, if $r_h(x_0) < 0$ for all the values of h but there is only one h corresponding to the maximum value $max_{h \in H} r_h(x_0)$, then, we can find a suitable constant C and modify R by means of $R_{mod}(x,u,y,h) = R(x,u,y,h) + C1_{\{x=x_0\}}$, for any x,u,y,h in such a way that for R_{mod} assumption (H1) holds true (and no change is introduced on the rewards for other values of x). Therefore, (H1) essentially means that for any x, there is only one value of h that maximizes $r_h(x)$. In practice, this is not a major restriction, since, again, this can be obtained by means of minor modifications of R and a fixed procedure to choose one h in case of "ties ".

Taking into account (H1), the following sets are well-defined

$$D_h = \{x \in S_X : r_h(x)0\},\ h = 0, ..., k.$$

and we have that

$$\bigcup_{h=0}^{k} D_h = S_X,\ D_h \cap D_l = \emptyset \text{ if } h \neq l.$$

The following lemma plays a key role in the rest of the paper. In two items of this lemma we will consider that T (size of the learning sequence) goes to infinity; since for the rest of the paper T will be fixed, we do not emphasize the dependence on T of $\hat{f}_j$, r_0, r_0^j.

We also denote

$$\gamma(x) := \sum_{f \in \mathcal{F}} \sum_{y \in S_Y} R(x, f(x), y, 0) p(y/x) P(\hat{f}_1 = f)$$

Lemma 3.1 *Let $\mathcal{F}$ be any class of functions from S_X on S_Y and let (X, Y) be a random vector, independent of the training sequence, distributed according to ρ. We have then that:*

- *i) For any j, $\lim_T \hat{f}_j(X) = f^{**}(X)$ in law*
- *ii) There exists a sequence of non-negative real numbers, $(a(T))_{T \in I\!N}$ such that $\lim_T a(T) = 0$ and, for any j,*

$$E\{\Big(r_0^j(X) - r_0(X)\Big)^2\} \leq a(T).$$

- *iii)Fix now T: for any x, $\lim_n \frac{1}{n} \sum_{j=1}^{n} r_0^j(x) = \gamma(x)$ a.s.*
- *iv) With the same notation as above,*

$$E\{(\gamma(X) - r_0(X))^2\} \leq a(T),$$

and for any $x \in S_X$,

$$|\gamma(x) - r_0(x)| \leq \left(\frac{a(T)}{\pi(x)}\right)^{\frac{1}{2}}.$$

Proof:

Fix j. Observe first that

$$\Gamma_T^j(f) - \Gamma(f) = \sum_{x \in S_X, y \in S_Y} R(x, f(x), y, 0)(\rho_T^j(x,y) - \rho(x,y)),$$

where

$$\rho_T^j(x,y) = \frac{1}{T} card\{i : 1 \leq i \leq T : X_i^j = x, Y_i^j = y\}.$$

Since $E\{\rho_T^j(x,y)\} = \rho(x,y)$ for any x, y, we deduce that

$$E\{\Gamma_T^j(f)\} = \Gamma(f). \tag{2}$$

In addition, by the Law of Large Numbers,

$$\lim_T |\rho_T^j(x,y) - \rho(x,y)| = 0 \ a.s., \text{ for any } j, x, y,$$

and, therefore

$$\lim_T max_{(x,y) \in S_X \times S_Y} |\rho_T^j(x,y) - \rho(x,y)| = 0 \ a.s.,$$

what implies in turn that

$$\lim_T max_{f \in \mathcal{F}} |\Gamma_T^j(f) - \Gamma(f)| = 0 \ a.s. \tag{3}$$

Let ω_0 be a point in the probability one set in which (3) holds. Define

$$C = \max_{f \in \mathcal{F}, f \neq f^{**}} \Gamma(f).$$

Let $\delta = \frac{1}{2}(\Gamma(f^{**} - C)$. From (3), there exists a natural number N (depending on ω_0) such that if $T \geq N$, then

$$\sup_{f \in \mathcal{F}} |\Gamma_T^j(f) - \Gamma(f)| < \delta$$

Therefore, if $T \geq N$ it must be $\hat{f}^j = f^{**}$ and (i) is proved.

From the previous argument, we also have that

$$\lim_T P(\hat{f}^j \neq f^{**}) = 0,$$

and, since R is bounded by (say) M,

$$E\{(r_0^j(X) - r_0(X))^2\} \leq (2M)^2 P(\hat{f}^j \neq f^{**})$$

which goes to zero with T and (ii) is proved.

For (iii), fix x. Then, $r_0^j(x)$, as said before, is a given function of the training sequence corresponding to the cycle j. Since training sequences of differents cycles are independent and follow the same distribution on the set of sequences of size T, we have that $r_0^1(x), ..., r_0^n(x), ...$ is an *iid* sequence, with mean

$$E\{r_0^j(x)\} = \sum_{f\in\mathcal{F}} \sum_{y\in S_Y} R(x, f(x), y, 0)p(y/x))P(\hat{f}_j = f)$$

Since the law of $\hat{f}_j$ does not depend on j we conclude that

$$E\{r_0^j(x)\} = \gamma(x)$$

and (iii) follows from the Law of Large Numbers.

Next, using (iii),

$$\begin{aligned} E\{(\gamma(X) - r_0(X))^2\} &= \lim_n E\{\left(\frac{1}{n}\sum_{j=1}^n (r_0^j(X) - r_0(X))\right)^2\} \\ &= \lim_n \|\frac{1}{n}\sum_{j=1}^n (r_0^j(X) - r_0(X))\|_{L^2}^2 \\ &\leq \lim_n \left(\frac{1}{n}\sum_{j=1}^n \|r_0^j(X) - r_0(X))\|_{L^2}\right)^2 \\ &\leq a(T) \text{ (by (ii))} \end{aligned}$$

Finally, pick any $x_0 \in S_X$ and write down

$$\begin{aligned} a(T) &\geq E\{(\gamma(X) - r_0(X))^2\} = \sum_{x\in S_X} (\gamma(x) - r_0(x))^2 \pi(x) \\ &\geq (\gamma(x_0) - r_0(x_0))^2 \pi(x_0) \end{aligned} \tag{4}$$

and we conclude that:

$$|\gamma(x_0) - r_0(x_0)| \leq \left(\frac{a(T)}{\pi(x_0)}\right)^{\frac{1}{2}} .\Diamond$$

We will also use in the sequel the following lemma, that is an easy consequence of the Law of Large Numbers for Martingales (see, for instance, [19]).

Lemma 3.2 *Assume that $(\mathcal{F}_j)_{j\in\mathbb{N}}$ is a filtration (i.e., each $\mathcal{F}_i$ is a sub-σ-algebra of the underlying σ-algebra $\mathcal{A}$ and, for any i, $\mathcal{F}_i \subset \mathcal{F}_{i+1}$) and that $\Delta_0, ..., \Delta_n, ...$ is a sequence of random variables such that:*

- *i)* Δ_i *is* $\mathcal{F}_{i+1}$*- measurable for any* i*.*
- *ii)*$E\{\Delta_i/\mathcal{F}_i\} = 0$ *for any* i*.*
- *iii) There exists* $K < \infty$ *such that* $sup_i|\Delta_i| \leq K$*, a.s.*

Then, if $M_n = \sum_{i=0}^{n-1} \Delta_i$*, we have that*

$$\lim_n \frac{M_n}{n} = 0 \ a.s.$$

Remark 3.3: The following fact also plays a key role in the proof of our main results. For any $x \in S_X$, define

$$\lambda_h(x) = r_h(x)1_{\{h>0\}} + \gamma(x)1_{\{h=0\}}.$$

Set

$$\Lambda_h = \{x \in S_X : \lambda_h(x)0\}.$$

It is clear that $\Lambda_h = D_h$ for $h > 0$. On the other hand, by Lemma 3.1 (iv), for T big enough, $\Lambda_0 = D_0$. More precisely, define

$$\eta = \min\{|r_0(x)| : x \in S_X\},\ T_0 = \inf\{T \in \mathbb{N} : \left(\frac{a(T)}{\pi(x)}\right)^{\frac{1}{2}} < \eta\ \forall x \in S_X\}.$$

Take $T \geq T_0$. If $x \in D_0$, then $r_0(x)\eta$ and by Lemma 3.1 (iv) and the definition of T_0, $\gamma(x)0$ and $x \in \Lambda_0$. If $x \notin D_0$, then $r_0(x) < -\eta$ and the same argument shows that $\gamma(x) < 0$, what implies that $x \notin \Lambda_0$. Therefore, if $T \geq T_0$,

$$\Lambda_h = D_h,\ h = 1, ..., 0.$$

We have then the first result, concerning the asymptotic behaviour of the credit matrix.

Theorem 3.1 *Let* T *be a fixed value,* $T \geq T_0$*, with* T_0 *as in Remark 3.3. As* n*, tends to infinity we have that*

$$\lim_n(\frac{1}{n}c_n(x,h))_{x\in S_X, h\in H} = (\lambda_h(x)\pi(x)1_{\{h=h(x)\}})_{x\in S_X, h\in H},\ a.s.$$

and

$$\lim_n h_n(x) = h(x)\ a.s.$$

(what implies that $h_n(x) = h(x)$ *for all* n *large enough, a.s.)*

Proof: Define

$$U_h^j = \{x \in S_X : h_j(x) = h\},$$

(set of points where the best advisor at cycle j is h)

$$\mathcal{T}_j = \sigma\left(\{(X_i^j, Y_i^j) : 1 \le i \le T\}\right)$$

(σ-algebra generated by the training sequence of cycle j),

$$\mathcal{V}_j = \sigma\left(\{(X_i^{v,j}, Y_i^{v,j}) : 1 \le i \le V\}\right)$$

(σ-algebra generated by the validation sequence of cycle j), and

$$\mathcal{F}_j = \bigvee_{i=1}^{j-1} (\mathcal{T}_i \vee \mathcal{V}_i)$$

(σ-algebra generated by training and validation sequences up to cycle $j-1$).

Set

$$\Delta_j = c_{j+1}(x,h) - c_j(x,h) - E\{c_{j+1}(x,h) - c_j(x,h)/\mathcal{F}_j\},$$

that clearly satisfies all the hypotheses of Lemma 3.2.
We have that:

$$\begin{aligned} c_{j+1}(x,h) - c_j(x,h) &= \frac{1}{V}\sum_{i=1}^{V} R(X_i^{v,j}, f_j(X_i^{v,j}), Y_i^{v,j}, h) 1_{\{X_i^{v,j}=x, h(X_i^{v,j})=h\}} \\ &= \frac{1}{V}\sum_{i=1}^{V} R(x, f_j(x), Y_i^{v,j}, h) 1_{\{X_i^{v,j}=x, x\in U_h^j\}}. \end{aligned}$$

We use in the following lines the fact that the validation sequence of cycle j is independent with respect to the training sample of cycle j and with respect to training and validation samples of previous cycles, and that U_h^j is $\mathcal{F}_j$-measurable. If $h > 0$ and $x \in U_h^j$, $f_j(x) = y_h(x)$ (deterministic) and

$$\begin{aligned} E\{c_{j+1}(x,h) - c_j(x,h)/\mathcal{F}_j\} &= E\{R(x, \hat{f}_j(x), Y_i^{v,j}, h) 1_{\{X_i^{v,j}=x\}} 1_{\{x\in U_h^j\}}\} \\ &= E\{R(x, y_h(x), Y_i^{v,j}, h) 1_{\{X_i^{v,j}=x\}}\} 1_{\{x\in U_h^j\}} \\ &= r_h(x)\pi(x) 1_{\{x\in U_h^j\}}. \end{aligned}$$

Therefore,

$$\Delta_j = c_{j+1}(x,h) - c_j(x,h) - r_h(x)\pi(x) 1_{\{x\in U_h^j\}} \text{ for } h \le k.$$

For $h = 0$, let us compute more carefully:

$$E\{c_{j+1}(x,0) - c_j(x,0)/\mathcal{F}_j\} = E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j\}1_{\{x\in U_0^j\}}.$$

But

$$\begin{aligned} E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j\} &= \\ E\{E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j \vee \mathcal{T}_j\}/\mathcal{F}_j\}. & \end{aligned}$$

Since $\hat{f}_j$ is $\mathcal{F}_j \vee \mathcal{T}_j$-measurable and $(X_i^{v,j}, Y_i^{v,j})$ is independent of $\mathcal{F}_j \vee \mathcal{T}_j$, we have that:

$$E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j \vee \mathcal{T}_j\} = \sum_{y\in S_Y} R(x,\hat{f}_j(x),y,0)p(y/x)\pi(x),$$

what implies in turn that:

$$E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j\} = \sum_{y\in S_Y} E\{R(x,\hat{f}_j(x),y,0)/\mathcal{F}_j\}p(y/x)\pi(x).$$

Observe now that $\hat{f}_j$ only depends on $\mathcal{T}_j$ and is independent of $\mathcal{F}_j$ (by its definition, $\hat{f}_j$ only depends on the performance of the elements of the model class $\mathcal{F}$ over the whole training sequence of cycle j), and thus,

$$\sum_{y\in S_Y} E\{R(x,\hat{f}_j(x),y,0)/\mathcal{F}_j\}p(y/x)\pi(x) =$$

$$\sum_{y\in S_Y}\sum_{f\in\mathcal{F}} R(x,f(x),y,0)P(\hat{f}_j = f)p(y/x)\pi(x).$$

Using now as in Lemma 3.1 the fact that the law of $\hat{f}_j$ does not depend on j, we conclude that

$$\begin{aligned} E\{R(x,\hat{f}_j(x),Y_i^{v,j},0)1_{\{X_i^{v,j}=x\}}/\mathcal{F}_j\} &= \\ \sum_{y\in S_Y}\sum_{f\in\mathcal{F}} R(x,f(x),y,0)P(\hat{f}_1 = f)p(y/x)\pi(x) &= \gamma(x). \end{aligned}$$

Therefore, for $h = 0$,

$$E\{c_{j+1}(x,0) - c_j(x,0)/\mathcal{F}_j\} = \gamma(x)1_{\{x\in U_0^j\}},$$

what shows that for any $h = 1, ..., k, 0$ we have

$$E\{c_{j+1}(x,h) - c_j(x,h)/\mathcal{F}_j\} = \lambda_h(x)1_{\{x\in U_h^j\}},$$

and that

$$\Delta_j = c_{j+1}(x,h) - c_j(x,h) - \lambda_h(x)\pi(x)1_{\{x\in U_h^j\}} \text{ for } h \leq 0.$$

Summing up both terms of this last equation with respect to j and dividing by n, we obtain as a consequence of Lemma 3.2 that:

$$\lim_n \left(\frac{c_n(x,h)}{n} - \lambda_h(x)\pi(x)\nu_n(h) \right) = 0, \ a.s.$$

where

$$\nu_n(h) = \frac{1}{n} card\{j : 0 \leq j \leq n-1, x \in U_h^j\}.$$

From now on, the rest of the proof is devoted to show the following two facts:

- a) $\lim_n \nu_n(h) = 1_{\{h=h(x)\}}$, for any x, h, and
- b) $\lim_n h_n(x) = h(x)$.

(Observe that b) it is not a direct consequence of a), since $\mu_n(h(x))$ gives only the asymptotic frequency of $h_n(x) = h(x)$).

To prove this, fix $x \in S_X$. Let ε be an arbitrary element of $(0,1)$. Set

$$a = inf_{h\in H}|\lambda_h(x)|\pi(x).$$

We already know that for almost any ω in our probability space, there exists $n_\varepsilon(\omega)$ such that

$$\max_{h\in H} |\frac{c_n(x,h)(\omega)}{n} - \lambda_h(x)\pi(x)\nu_n(h)| < \frac{1}{2}a\varepsilon \tag{5}$$

for any $n \geq n_\varepsilon(\omega)$.

Fix ω as before. Let us assume for a moment that:

$$\text{There exists } n_1 \geq n_\varepsilon(\omega) \text{ such that } \nu_{n_1}(h(x))(\omega) \geq \varepsilon. \tag{6}$$

By Remark 3.4, we know that $\Lambda_h = D_h$ for any h. Hence, $\lambda_h(x)0$ if and only if $h = h(x)$. We have then that:

$$\frac{c_{n_1}(x,h(x))(\omega)}{n_1} - \lambda_{h(x)}(x)\pi(x)\nu_{n_1}(h(x))(\omega) - \frac{a\epsilon}{2} \quad \geq$$

$$\lambda_h(x)\pi(x)\nu_{n_1}(h)(\omega) + \frac{a\epsilon}{2} > \frac{c_{n_1}(x,h)}{n_1}(\omega) \qquad \text{for any } h \neq h(x)$$

This implies that $h_{n_1}(x)(\omega) = h(x)$ and hence,

$$\nu_{n_1+1}(h(x))(\omega) = \frac{n_1\nu_{n_1}(h(x))(\omega) + 1}{n_1 + 1}\varepsilon.$$

Therefore, the same argument may be applied to $n_1 + 1$ instead of n_1 and we conclude that

$$h_n(x)(\omega) = h(x) \text{ for any } n \geq n_1$$

what clearly implies

$$\lim_n \nu_n(h)(\omega) = 1_{\{h=h(x)\}}(\omega), \text{ for any } h.$$

It is enough now to show that, on a set of probability one, there exists $\varepsilon \in (0, 1)$ such that (6) holds true.

Let us call A to subset of Ω where (6) does not hold for any $\varepsilon \in (0, 1)$. It is clear that

$$A = \{\omega \in \Omega : \lim_n \nu_n(h(x))(\omega) = 0\}.$$

We will prove that $P(A) = 0$. Observe that the reward function R is bounded (indeed, its domain is a finite set) and hence $\frac{c_n(x,h)}{n}$ is bounded, allowing to interchange limits and expectations in the following lines.
By the definition of $h_n(x)$ we have that, for any ω in Ω and h in H,

$$\frac{c_n(x, h_n(x))}{n}(\omega) \geq \frac{c_n(x, h)}{n}(\omega)$$

what implies that for any h,

$$E\{1_A \frac{c_n(x, h_n(x))}{n}\} \geq E\{1_A \frac{c_n(x, h)}{n}\} \tag{7}$$

Using that

$$\lim_n \max_{h\in H} |\frac{c_n(x, h)}{n} - \lambda_h(x)\pi(x)\nu_n(h)| = 0 \ a.s.$$

and taking limits in (7) we deduce that, for any h,

$$\limsup_n E\{1_A\lambda_{h_n(x)}(x)\pi(x)\nu_n(h_n(x))\} \geq \limsup_n E(1_A\lambda_h(x)\pi(x)\nu_n(h)\}.$$

Since the right-hand side of the last inequality is non-negative for $h = h(x)$, so is the left-hand side, hence

$$\limsup_n E\{1_A\lambda_{h_n(x)}(x)\pi(x)\nu_n(h_n(x))\} \geq 0.$$

But if we take

$$a(x) = \max_{h \neq h(x)} r_h(x),$$

which is negative, it is easy to check that the left-hand side of the last inequality is smaller than

$$\limsup_n E(1_A a(x)\pi(x)1_{\{h_n(x)\neq h(x)\}}) \tag{8}$$

and therefore (8) must be non-negative.

We will show that (8) is negative if $P(A)$ is greater than zero, leading to a contradiction. Taking into account that $a(x)$ is negative, using Fatou's lemma for negative functions and the fact that

$$\limsup_n 1_{\{h_n(x)\neq h(x)\}} = 1 \text{ over } A$$

we conclude that (8) is smaller than

$$E(1_A a(x)\pi(x)),$$

which is negative if $P(A) > 0$.$\Diamond$

Remark 3.4: It must be noticed that in the previous result we have assumed that T is big enough (i.e., $T \geq T_0$), but fixed.

Next result shows that a CLT holds for the convergence of Theorem 3.1. It is based on the following version of the CLT for martingales (see [19]).

Lemma 3.3 *Under the assumptions of Lemma 3.2, if in addition there exists a non-negative constant σ^2 such that*

$$\lim_n \frac{1}{n}\sum_{i=0}^{n-1} E\{\Delta_i^2/\mathcal{F}_i\} = \sigma^2 \text{ in probability,}$$

then

$$\lim_n \frac{1}{\sqrt{n}} M_n = N(0,\sigma^2) \text{ in law .}$$

Remark 3.5: A straightforward argument gives the multivariate version of Lemma 3.3., that may be stated as follows. Assume that $(\Delta_i(1),...,\Delta_i(d))_{i\in\mathbb{N}}$, is a d-dimensional sequence such that $(\Delta_i(s))_{i\in\mathbb{N}}$ satisfies the assumptions of Lemma 3.2 for each $s = 1,...,d$ with respect to the same filtration $(\mathcal{F}_i)_{i\in\mathbb{N}}$ and that there exists a covariance matrix M such that

$$\lim_n \frac{1}{n}\sum_{i=0}^{n-1} E\{\Delta_i(s)\Delta_i(t)/\mathcal{F}_i\} = M(s,t) \text{ in probability, for any } s,t = 1,...,d.$$

Then if $M_n = (M_n(1), ..., M_n(d))$ is defined by $M_n(s) = \frac{1}{n}\sum_{i=0}^{n-1} \Delta_i(s)$, we have that

$$\lim_n \frac{1}{\sqrt{n}} M_n = N(0, M) \text{ in law.}$$

where $N(0, M)$ denotes a d-dimensional centered gaussian random vector with covariance matrix M.

Theorem 3.2 *We have that*

$$\lim_n \sqrt{n} \left(\frac{1}{n} c_n(x,h) - \lambda_h(x)\pi(x) 1_{\{h=h(x)\}} \right)_{x \in S_X, h \in H} = N(0, M) \text{ in law.}$$

where, for any $x, x^* \in S_X,\ h, h^* \in H$,

$$M(x,h;x^*,h^*) = \frac{V-1}{V} \lambda_h(x)\lambda_{h^*}(x^*)\pi(x)\pi(x^*) 1_{\{h(x)=h, h(x^*)=h^*\}},$$

$$-\frac{1}{V}\theta_h(x)\pi(x) 1_{\{x=x^*, h=h^*=h(x)\}},$$

and

$\theta_h(x) := E\{R(x, y_h(x), Y, h)^2\}$ *for* $h = 1, ..., k;\ \theta_0(x) := E\{R(x, \hat{f}^1(x), Y, h)^2\}$.

Proof: Set

$$\Delta_j(x,h) = c_{j+1}(x,h) - c_j(x,h) - \lambda_h(x)\pi(x) 1_{\{x \in U_h^j\}} \text{ for } x \in S_X, h \in H.$$

After Remark 3.5, it is clear that is suffices to prove the following facts:

- a) $\lim_n \frac{1}{n}\sum_{j=1}^n E\{\Delta_j(x,h)\Delta_j(x^*,h^*)/\mathcal{F}_j\} = M(x,h;x^*,h^*)$ in probability, for any $x, x^* \in S_X,\ h, h^* \in H$.
- b)$\lim_n \sqrt{n}\left(\frac{1}{n}\sum_{j=0}^{n-1} 1_{\{x \in U_h^j\}} - 1_{\{h=h(x)\}}\right) = 0$ *a.s.* for any $x \in S_X$, $h \in H$.

By Theorem 3.1., for each $x \in S_X$, and for any ω on a set of total probability, there exists $n(\omega) \in \mathbb{N}$ such that for any $n \geq n(\omega)$ we have $h_n(x)(\omega) = h(x)$. It is then clear that, for $n \geq n(\omega)$, we have that

$$|\frac{1}{n}\sum_{j=0}^{n-1} 1_{\{x \in U_h^j\}}(\omega) - 1_{\{h=h(x)\}}| \leq \frac{n(\omega)}{n},$$

what clearly implies b).

We will then focus on a). It is easy to check that

$$E\{\Delta_j(x,h)\Delta_j(x^*,h^*)/\mathcal{F}_j\} = E\{(c_{j+1}(x,h)-c_j(x,h))(c_{j+1}(x^*,h^*)-c_j(x^*,h^*)\}$$

$$-\lambda_h(x)\lambda_{h^*}(x^*)\pi(x)\pi(x^*)P(h_j(x)=h, h_j(x^*)=h^*).$$

But

$$E\{(c_{j+1}(x,h)-c_j(x,h))(c_{j+1}(x^*,h^*)-c_j(x^*,h^*))\} =$$

$$\frac{1}{V^2}\sum_{s=1}^{V}\sum_{t=1}^{V} E\{R(x,f_j(x),Y_s^{v,j},h)R(x^*,f_j(x^*),Y_t^{v,j},h^*)1_{\{X_s,x,h\}}1_{\{X_s,x^*,h^*\}}\}$$

where $1_{\{X_s,x,h\}} := 1_{\{X_s^{v,j}=x,h_j(x)=h\}}$ and $1_{\{X_s,x^*,h^*\}} := 1_{\{X_t^{v,j}=x^*,h_j(x^*)=h^*\}}$.

If $s \neq t$, then

$$E\{R(x,f_j(x),Y_s^{v,j},h)R(x^*,f_j(x^*),Y_t^{v,j},h^*)1_{\{X_s,x,h\}}1_{\{X_s,x^*,h^*\}}\} =$$

$$\lambda_h(x)\lambda_{h^*}(x^*)\pi(x)\pi(x^*)P(h_j(x)=h, h_j(x^*)=h^*).$$

On the other hand, if $s = t$, then

$$E\{R(x,f_j(x),Y_s^{v,j},h)R(x^*,f_j(x^*),Y_t^{v,j},h^*)1_{\{X_s,x,h\}}1_{\{X_s,x^*,h^*\}}\} =$$

$$\theta_h(x)\pi(x)1_{\{x=x^*,h=h^*\}}P(h_j(x)=h).$$

Therefore, we have that

$$E\{(c_{j+1}(x,h)-c_j(x,h))(c_{j+1}(x^*,h^*)-c_j(x^*,h^*))\} =$$

$$\frac{(V-1)}{V}\lambda_h(x)\lambda_{h^*}(x^*)\pi(x)\pi(x^*)P(h_j(x)=h, h_j(x^*)=h^*)$$

$$-\frac{1}{V}\theta_h(x)\pi(x)1_{\{x=x^*,h=h^*\}}P(h_j(x)=h),$$

and applying Theorem 3.1, a) follows easily.◊

Remark 3.6: Observe that for each pair x, x^*, the limit covariance matrix is null except in the case $h(x)=h, h(x^*)=h^*$. Indeed, instead of the whole credit matrix, we may consider the reduced mean credit vector $(\frac{1}{n}c_n(x,h(x)))_{x\in S_X}$ since no other term is relevant for the asymptotic behaviour of the algorithm.

4 RRLA vs NRLA.

Next, we give the asymptotic behavior of the credit matrix when the NRLA algorithm is used.

Theorem 4.1 *For NRLA algorithm, we have that*

$$\lim_n(\frac{1}{n}c_n(x,h))_{x\in S_X,h\in H} = (\lambda_h(x)\pi(x)1_{\{h=h(x)\}})_{x\in S_X,h\in H},\ a.s.$$

Proof: For the sake of clearity, let us use a different notation for the principal ingredients of the algorithm in the NRLA case. Let us now denote $\hat{h}_j(x)$ the analogous of $h_j(x)$, g_j the analogous of f_j and $\hat{g}^j$ the analogous of $\hat{f}^j$. More precisely, we assume now that, at cycle j, the available training sample is

$$(X_i^s, Y_i^s)_{1\leq i\leq T, 1\leq s\leq j}.$$

Hence, from the model we choose $\hat{g}^j$ such that

$$\hat{g}^j = \text{argmax}_{f\in\mathcal{F}}\,\Gamma_{jT}(f)$$

where

$$\Gamma_{jT}(f) = \frac{1}{jT}\sum_{s=1}^{j}\sum_{i=1}^{T} R(X_i^s, f(X_i^s), Y_i^s, 0).$$

Finally, $\hat{h}_j(x)$ is now the most credible advisor among tke k experts and $\hat{g}^j$, the credit matrix is updated exactly as before (i.e., using only the validation sequence correspondig to each cycle) and g_j denotes the predictor.

If we now set

$$\hat{r}_0^j(x) = \sum_{y\in S_Y} R(x, \hat{g}^j(x), y, 0)p(y/x),$$

it is clear that

$$\left(\hat{r}_0^j(x)\right)_{x\in S_X}$$

has the same law as, in the RRLA,

$$\left(r_0^j(x)\right)_{x\in S_X}$$

when a training sample of size jT is used, and therefore, by Lemma 3.1 ii), it converges in L^2, as j goes to infinity, to $r_0(x)$.

From now on, the proof follows very closely the arguments used in Theorem 3.1 and may be easily reproduced by the reader. $\Diamond$

As a direct consequence of Theorem 3.1 and 3.2, we can finally compare the performance of RRLA and NRLA. We will compare performances by means of the following performance ratio:

$$\tau_j := \frac{E\{R(X, f_j(X), Y, h_j(X))\}}{E\{R(X, g_j(X), Y, \hat{h}_j(X))\}}.$$

We have then

Theorem 4.2 *If in RRLA we use $T \geq T_0$, then*

$$\lim_j \tau_j = \frac{E\{\lambda_{h(X)}\}}{E\{r_{h(X)}\}} = 1 - \frac{E\{(r_0(X) - \gamma(X))1_{\{X \in D_0\}}\}}{E\{r_{h(X)}\}}.$$

Proof: As seen before, we have that

$$E\{R(X, f_j(X), Y, h_j(X))\} =$$

$$\sum_{h=0}^{k} \sum_{x \in S_X} \sum_{y \in S_Y} R(x, y_h(x), y, h) P(h_j(x) = h) p(y/x) \pi(x) +$$

$$\sum_{x \in S_X} \sum_{f \in \mathcal{F}} \sum_{y \in S_Y} R(x, f(x), y, h) P(\hat{f}^j = f) P(h_j(x) = 0) p(y/x) \pi(x) =$$

$$\sum_{h=0}^{k} \sum_{x \in S_X} r_h(x) \pi(x) P(h_j(x) = x) + \sum_{x \in S_X} \gamma(x) \pi(x) P(h_j(x) = 0) =$$

$$\sum_{h=0}^{k} \sum_{x \in S_X} \lambda_h(x) \pi(x) P(h_j(x) = h).$$

By Theorem 3.1,

$$\lim_j P\{h_j(x) = h\} = 1_{\{h = h(x)\}},$$

and therefore,

$$\lim_j E\{R(X, f_j(X), Y, h_j(X))\} = \sum_{h=0}^{k} \sum_{x \in S_X} \lambda_h(x) \pi(x) 1_{\{h = h(x)\}} =$$

$$\sum_{x \in S_X} \lambda_{h(x)}(x) \pi(x) = E\{\lambda_{h(X)}(X)\}.$$

In a similar way, using Theorem 3.2, we deduce that

$$\lim_j E\{R(X, f_j(X), Y, h_j(X))\} = E\{r_{h(X)}(X)\}.$$

Finally, observe that

$$E\{\lambda_{h(X)}(X)\} = E\{\lambda_{h(X)}(X)1_{\{X\notin D_0\}}\} + E\{\lambda_0(X)1_{\{X\in D_0\}}\} = E\{r_{h(X)}(X)1_{\{X\notin D_0\}}\} + E\{\gamma(X)1_{\{X\in D_0\}}\},$$

and the result follows. $\Diamond$

The following corollary illustrates on the applications of Theorem 3.4.

Corollary 4.1

i) Under the assumptions of Theorem 3.4, we have

$$\lim_j \tau_j \geq 1 - \frac{(a(T)\pi(D_0))^{\frac{1}{2}}}{E\{r_{h(X)}\}}.$$

ii) Set $A = \min_{x\in D_0} r_0(x), B = \min_{x\notin D_0} r_{h(x)}(x)$.
Then

$$\lim_j \tau_j \geq 1 - \frac{(a(T)\pi(D_0))^{\frac{1}{2}}}{A\pi(D_0) + B(1-\pi(D_0))}.$$

Proof:

To prove i), observe that $E\{r_{h(X)}(X)\}$ is positive and apply Cauchy-Schwarz inequality and Lemma 3.1 iv) in Theorem 3.4.

To prove ii), observe that

$$E\{r_{h(X)}(X)\} \geq A\pi(D_0) + B\left(1-\pi(D_0)\right).\Diamond$$

Remark 4.7: Corollary 4.1 says that if T is large, both algorithms have very similar performances. On the other hand, if $T \geq T_0$ but T is not very large, the ratio of performances is close to one if $\pi(D_0)$ is small or if A or B are large. In simple words, this means that performance is almost the same if training samples are large, or experts are the most credible advisor for almost all inputs, or experts have a very good mean performance (even if they are not almost allways chosen as the best advisor) or the model has a very good mean performance. In other words if RRLA is not close in performance to NRLA, we have awfully chosen the ingredients of our learning machine! In [12] very impressive numerical examples of performances of both algorithms are given.

Acknowledgements: to an anonymous referee for his very precise and valuable suggestions.

References

[1] Aspirot L., Belzarena P., Bermolen P., Ferragut A., Perera G., Simón M. (2005). Quality of Service Parameters and Link Operating Point Estimation Based on Effective Bandwith. *Performance Evaluation* 59, 103-120.

[2] Aspirot L., Belzarena P., Perera G., Bazzano B. (2005) End-To-End Quality of Service Prediction Based On Functional Regression. *HET-NET 2005*, P46 (also available in http://iie.fing.edu.uy/investigacion/grupos/artes/).

[3] Bel L., Bellanger L., Bonneau V., Ciuperca G., Dacunha-Castelle D., Deniau C., Ghattas B., Misiti Y., Oppenheim G., Poggi J.M., Tomassone R.(1999). Elments de comparaison de prvisions statistiques des pics d'ozone. *Revue de Statistique Applique*, vol. XLVII (3), 7-25.

[4] Belzarena P., Bermolen P., Casas P., Simón M. (2005). Virtual Path Networks Fast Performance Analysis.(to apper in *Performance Evaluation*).

[5] Bolton R.J., Hand D.J.(2002). Statistical Fraude Detection: A Review (with discussants). *Statist. Sci.* Vol 17, No. 3, 235-255.

[6] Buschiazzo D., Ferragut A., Vázquez A., Belzarena P. (2005). Fast Overflow Probability Estimation Tool for MPLS Networks. *LANC 2005, Session 5 (MPLS)* (also available in

http://iie.fing.edu.uy/investigacion/grupos/artes/).

[7] Bacceli F., Bolot J., Machiraju S., Nucci A., Veitch D. (2005). Theory and Practice of Cross-Traffic Estimation. *SIGMETRICS* '05 June 6-10, 2005, Barnff,Alberta, Canada.

[8] Breiman L., Friedman J.H.,Olshen R.A., Stone C.J. (1999). *Classification And Regression Trees* CA.

[9] Cucker F., Smale S. (2002). *On the mathematical foundations of learning.* Bulletin of the American Mathematical Society, Vol. 39, N. 1, p. 1-49.

[10] Devroye L., Gyorfi L., Lugosi G. (1996) *A Probabilistic theory of Pattern Recognition.* Springer.

[11] Flash P.A. (2000). *On the state of the art in Machine Learning: a personal review.* Department of Computer Science, University of Bristol.

[12] Ghattas B., Perera G. (2005). A Resource-Restricted Learning Algorithm for on-line evaluation of non-stationary data. *Submitted.*

[13] Hastie T., Tibshirani R., Friedman J. (2001) *The Elements of Statistical Learning. Data Mining, Inference, and Prediction.* Springer .

[14] Karagiannis T., Molle M., Faloutsos M. and Broido A. (2004). A Nonstationary Poisson View of Internet Traffic. *IEEE INFOCOM 2004*.

[15] Kelly F. (1996). Notes on Effective Bandwith. *Stochastic Networks, Theory and Applications*, edited by Kelly, Zachiaris and Ziedis, Oxford University Press.

[16] Pechiar J., Perera G., Simón M. (2001) *Effective bandwith estimation and testing for Markov sources.*, Performance Evaluation 945,p. 1-19.

[17] Smale S. (2000) Mathematical problems for the next century. Arnold, V. (ed.) et al., *Mathematics: Frontiers and perspectives.* Providence, RI: American Mathematical Society (AMS). 271-294.

[18] Vapnik V. (1998) *Statistical Learning Theory.* Wiley.

[19] Williams D. (1991) *Probability with martingales.* Cambridge University Press.

[20] Zhang Y., Duffield N., Paxson V. and Shenker S. (2001). On The Constancy of Internet Path Properties. *ACM SIGCOMM Internet Measurement Workshop.*

Institut de Mathématiques de Luminy, CNRS, Marseille, France
Universidad de la República, Montevideo, Uruguay
ghattas@lumimath.univ-mrs.fr, gperera@fing.edu.uy