

Multi One-Class Incremental SVM For Both Stationary And Non-Stationary Environment

Anh-Khoi Ngo-Ho^{*1}, Nicolas Ragot^{†1}, Veronique Eglin^{‡2}, et Jean-Yves Ramel^{§1}

¹Laboratoire d'Informatique (LI), Université François Rabelais Tours, France

²Laboratoire d'InfoRmatique en Image et Systemes d'information (LIRIS), INSA de Lyon, France.

1^{er} octobre 2014

Résumé

In previous works, a multi One-Class incremental SVM (mOC-iSVM) system has been proposed as a solution to learning problems in a context of data streams. In the field of incremental learning, different challenges are addressed to adapt the system to stationary (fixed concepts) or non-stationary environments (concept drift). The mOC-iSVM has shown its ability to improve the classification accuracy over the time, with incoming new labeled data, without performing any complete retraining. The principle lies on the one-class SVM method combined with incremental process that preserves old support vectors (SVs). The paper presents an adaptation to stationary and non-stationary environment with a support vector age pruning. This is done thanks to the history knowledge of models that have been used in the past by a selection of old support vectors according to the age they contribute in the history. This new design is called multi One-Class incremental SVM with Age Pruning (mOC-iSVM.AP) and its performances are evaluated on two frequently tested benchmarks on both sudden (abrupt, instantaneous) drift and gradual drift with recursive environment (SEA and Checker Board (CB) dataset) with quite promising results and perspectives.

Mots-clef : incremental learning, learning in non-stationary environments, concept drift, One-Class SVM.

*ngoho@univ-tours.fr

†nicolas.ragot@univ-tours.fr

‡veronique.eglin@insa-lyon.fr

§jean-yves.ramel@univ-tours.fr

1 Introduction

One of the most difficult problems with classification is that the concepts and their relations are highly dependent on some hidden context. As consequences, any information about number of concept, description of concepts, relation between the concepts should ideally be totally well-known. Most often, a priori or arbitrary knowledge is injected in the learning scheme to simplify the task and make it more accurate. This is the reason why many supervised classification tasks were done under the assumption that the contexts are fixed and known before the learning. Recently, a great attention has been paid on classification in data streams where the concepts can change over the time. The classification of data delivered in stream mainly use very specific techniques and adaptations. Indeed, for many real processes, the categorization of the data is not so simple, even in a supervised mode. Each time the user encounters a new data, he can decide to change the previously established classification, for example by merging classes, creating new ones and so on. More generally, the concepts can evolve over the time depending on encountered data : it can lead to concepts fusion or splitting, to concepts creation, extension or deletion. The knowledge and expertise are not precisely accessible at the beginning of the process mainly because final needs and usages can vary.

Some specific cases of dynamic problems have been studied in the literature under the following terms : "concept drift", "incremental learning", "decremental learning", and "learning in non-stationary environment" ([?], [?], [?], [?], [?], [?], [?] [?] [?], [?], [?], [?], [?], and [?]). Polikar's works are probably the most significant contributions to this field. But it is not clearly

established in the literature whether these approaches can perform on most of dynamic situations that can be encountered in data streams. In this paper, we intend to show the ability of our mOC-iSVM.AP to be well suited for data classification in situations of evolution (concept drift) through time. The goal is not to prove that the average accuracy of our proposal outperforms others (like proposed in [?] and [?]) but the main idea is to show that mOC-iSVM.AP is very efficient in a large variety of dynamic situations on both stable and non-stable environment.

The paper is organized as follows. First, we present the related work on incremental learning and learning in non-stationary environment. The advantages and the disadvantages of mOC are discussed and a new version based on age pruning is described. It is called mOC-iSVM.AP for multi One-Class incremental SVM with Age Pruning. In the experimental last section, different experiments are detailed with the comparisons with others state-of-the-art performant methods.

2 Related Works

In dynamic environments, The first category of systems is related to classification over the time with an increasing number of classes. The second category is related to classification with an adaptation to concepts that can change through time ("concept drift", "decremental learning", "learning in non-stationary environment").

The term "Incremental learning" (that has been widely used in the literature but for a great variety of situations) has been more precisely defined in [?] with four criteria : it should be able to learn additional information from new data ; it should not require to access to the original data ; it is used to update an existing classifier ; it should preserve previously acquired knowledge (it should not suffer from catastrophic forgetting) ; and it should be able to learn the classes when they are introduced with new data. We can make reference to [?] based on ARTMAP (Adaptive resonance theory) ; [?] based on SOM (Seft Organization Maps) ; [?] based on FIS (Fuzzy Inference System). For these approaches, the main idea is to model a new class by adding a new prototype and adapting the parameters of the decision rules. Learn++ [?], is based on ensemble of classifiers in which a new class is handled by adding new classifiers with a weighted voting scheme.

An other type of evolution and adaptation in dynamic systems corresponds to a learning in "non stationary environment" and "concept drift". They represent

the capacity of system to adapt its context to concepts changes over time. This drift can be an extension, a reduction or a translation of concepts (by forgetting old parts of the concept for example). The best adaptation can be realized by selecting examples used in each training step, or by keeping a fixed number of new instances while keeping some old knowledge with a "sliding window" mechanism as explained in [?]. It is also possible to weight the incoming examples as in [?]. Approaches based on ensemble of classifiers have been implemented for improving the adaptation. Other mechanisms privilege the creation of new classifiers for each new dataset and delete the oldest one with a "pruning method" as in [?]. Finally some techniques lie on weighting mechanisms relative to new incoming data, as proposed in [?]. All these changes can be performed with online bagging and boosting in [?].

More recently, some authors proposed the combination between incremental learning and concept drift. Elwell in [?] combines incremental learning Learn++ and "weighting on newest" approach to adapt to concept drift. Bouillon in [?] also introduces "decremental learning" with incremental learning ability of FIS and and a concept drift approach based on a "sliding window" (only few old data are considered relative to the temporal sliding window). We proposed in [?] a multi one-class incremental SVM (mOC-iSVM) that can handle both incremental learning and learning in non-stationary environment. The main difficulties when a classifier has to deal with evolving concepts comes from the inter-dependency of the parameters, especially in discriminant approaches like [?] of Bouillon, since they determine decision boundaries relatively to several classes. As a consequence, when a concept is changed, all parameters have to be adapted, and require many examples for training. This constraint is contradictory to a classification in a context of data streams. This is the reason why Elwell in [?] is one of the most promising current approach with an ensemble of classifiers that split the overall problem into many sub-problems processed by expert dedicated classifiers. Nevertheless, a strong relationship remains between classifiers and there could be a huge redundancy of information inside the set. In [?], we take advantage of the independencies that leads naturally into one-class classifiers that only use positive data from one class to build their model. It could have a huge potential : insensibility to imbalanced data ; rejection process to detect new concepts or ambiguities (and thus potential merging of concepts) ; multiclass labelling ; individual feature space, etc. But it is also known to be a less accurate than discriminative approaches [?]. And it has

also a complex process of parameter selection that is not easy to handle [?].

In the next session, we will talk about the advantages and disadvantages of mOC-iSVM and propose a new version of mOC-iSVM called multi One-Class incremental SVM with Age Pruning (mOC-iSVM.AP) that improve mOC-iSVM both in stable and non-stable environment.

3 Multi One-Class Incremental SVM With Age Pruning (mOC-iSVM.AP)

As mentioned above, the variety of potentials of the one-class SVM incremental mechanism has convinced us to use it for dynamic environment classification. In [?], the design of a multi One Class incremental SVM method (mOC-iSVM) has been proposed to resolve the learning and recognition stages of a multi-class classification problem. In the proposition, each one-class SVM is trained incrementally at each step, using the Syed procedure [?], i.e. using the classic SVM learning procedure with old support vectors embedded with new incoming data having the same label than than the class modeled by the SVM.

One difficulty with this kind of system is that the stability and the plasticity of incremental SVM are very sensitive to the ratio existing between the old knowledge depicted by old support vectors and new knowledge associated to new data. The choices for the system parameterization associated to the size of the learning packet for the One-Class SVM determine those very influential relations. The more we keep knowledge, the more the system is stable, and by inverse, the more the new knowledge is kept, the more the system is considered as dynamic. In addition, in one-class SVM, the parameters of system, especially the parameter ν (pronounced *nu*) that approximates the fraction of training errors over all selected support vectors, are very decisive factors for establishing the generality or the specificity of the model. The SVM model is also largely dependent on the selection of the model parameters (selection of the kernel's parameters and the soft margin parameter). In some cases, some parameters can be more adapted for stability/plasticity problem, and inversely. The choice of the best compromise is mainly solved by the automatic grid search procedure during training.

As other advantage associated to One-class SVM, we can notice its ability to consider "out-of-date" support

vectors for a given class even if new incoming data arrive (for the same considered class). As a consequence, the "long time memory" of the system can be strengthened in stable environment. Inversely, the immediate adaptation to an unstable environment (with sudden evolution) can be slowed down.

Considering the two challenges above, the second one is more complex because it also depends both on environment stability and size of incoming new data. In mOC-iSVM [2], there is no possibility for the operator to precisely control the efficiency and support vectors quantity : this task is dedicated to the procedure of parameters selection. In the situations where the environment properties (stability or plasticity) are unknown, the mOC-iSVM is an interesting choice. No a priori knowledge on the environment guides the system to make the optimal choices in considering both ancient SVs (from previous steps) and the last selected ones (from the more recent). Otherwise when the environment is well known, mOC-iSVM does not provide any satisfying solution to improve the SVs selection, because the selection lies on the same procedure (with both old and new SVs). Consequently, a lack of adaptation can be deplored. By considering the plasticity/stability dilemma, we need a simple method that can adapt mOC-iSVM in both stable and non-stable environments. The solution is proposed through a new version of mOC-iSVM with a SVs selection based on age pruning. We have been influenced by Wang's work in [?]. The idea is simple : for each selected SV, we keep its age corresponding to the time it appears, and we control the plasticity/stability of the system via this age structure. The age hierarchy can be designed as follow : among the new incoming data (in the new lot to be processed), a selection of new support vectors is done with a corresponding age that will be set to 1. Consequently, at each step, the older lots of support vectors have their age increased by 1 unit. An *age* parameter α (corresponding to a mechanism based on age pruning) has to be set before the learning and only support vectors with a lower age than α (so being more recent) can be used for the new model training. Within that principle, other support vectors (having higher ages than α) can be kept if necessary by the system for a future usage.

Reported to our proposition, if the only youngest SVs are used in the system ($\alpha = 1$), the mOC-iSVM.AP is more dynamic (adaptation of the new context with embedded SVs from the just previous one)- see figure 1, and inversely, if all SVs (from the youngest to the oldest ones with the maximal age) are used in the system, the mOC-iSVM.AP becomes more efficient in stable

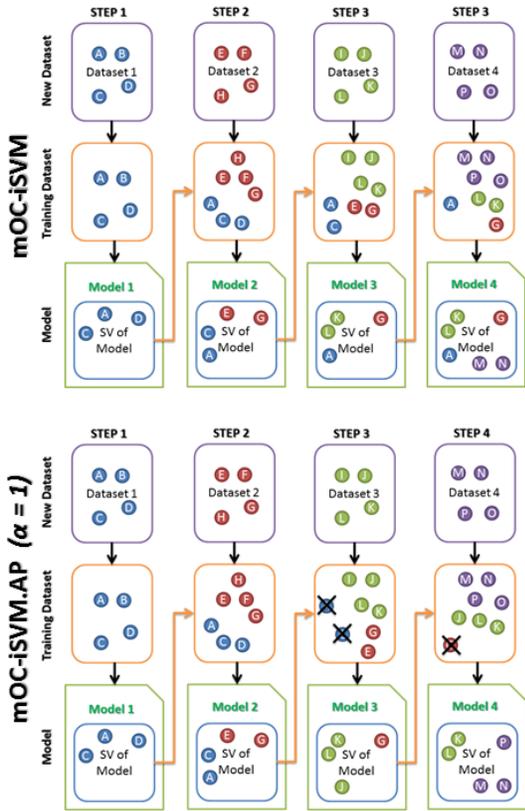


FIGURE 1 – Differences between mOC-iSVM and mOC-iSVM.AP ($\alpha=1$)

environments (the SVs of each One Class SVM have a better control of the frontiers). By using a selection of the very recent SVs only, mOC-iSVM.AP limits the influence of oldest (and “out-of-date”) information and consequently speeds up its reactions : any environment change will not need more time to be integrated. By taking into account the newest important SVs to describe the concept, the system can manage more easily and with a high plasticity any concept drift. By considering only recent SVs while new incoming data arrive, we ensure that no out-of-date information can affect the actual model. The limitation of old knowledge can improve the plasticity of the system. All questions relative to the generality *vs* specificity of the system are considered as previous.

The automatic selection of α has also been studied (by considering the models performances) but it is not included in this paper because of space limitation. In any cases, we just used the basic a priori knowledge : if the datasets are unstable with quick changes, $\alpha=1$ (figure 1) ; if they are stable, $\alpha=\text{max-value}$. In this work

we don’t try to find the best value for α (it will be considered in the perspectives of this work). Even with such simple heuristic, the results are better (other parameters are tuned either automatically by grid-search or empirically in order to maximize the performances of both approaches). By considering age pruning, the influence of old data is still considered by the only new resulting SVs are preserved (see figure 1).

4 Experiments In Non-Stationary Environment

In this part, we intend to show the ability of our mOC-iSVM.AP system to adapt the recognition to non-stationary environment. The ability of mOC-iSVM has been initially demonstrated for stationary environment in [?] ¹. We show here how mOC-iSVM.AP can outperform mOC-iSVM on the non-stationary environment.

Firstly, we describe the two datasets and the protocol used for the performance evaluation and for the discussion with the best incremental approaches of the state-of-the-art. After a brief discussion of the weakness of mOC-iSVM, we show how mOC-iSVM.AP can improve its behavior and outperform some of the best incremental mechanisms for non-stationary environment. Some improvements and perspectives for a still better adaptation to environment are also presented.

4.1 SEA and Rotating Checkerboard Dataset

The SEA Dataset is the proposed dataset on which the original SEA (Streaming Ensemble Algorithm) has been developed and tested by Street in [?]. This dataset is considered as a standard benchmark for concept drift study. The dataset includes two classes and three features. It is characterized by periods that introduce new data with occasional sharp changes in the class boundaries, i.e., sudden concept change. Class labels are assigned based on the sum of the relevant features : an instance is assigned to class 1 if the sum of the two first features falls below a predefined threshold, and otherwise it is assigned to class 2. The threshold has been changed three times ($8 \rightarrow 9 \rightarrow 7.5 \rightarrow 9.5$) creating an abrupt shift in the class boundary. Each step contains 250 instances for training and the same quantity for testing in totally 200 steps. In SEA Dataset,

¹. That is the reason why only a short experiment on stationary environment was done with mOC-iSVM.AP in this paper.

we evaluate mOC-iSVM.AP with the protocol proposed by L++.NSE in [?]. In addition, we also test our mOC-iSVM.AP with two others size of incoming data (500 and 1000 instead of 250 for each step), considering the impact (previously discussed) of the size of the incoming packets.

Rotating Checkerboard (CB) Dataset [?] : The dataset derived from a canonical XOR problem which will undergo variable rate drifts during time : constant drift rate, pulse drift rate, exponentially increasing drift rate, and sinusoidal drift rate. The angle and location of the decision boundaries change drastically at every step . In order to prevent training on identical snapshots of data and to increase complexity, 10% of random noise has been introduced. Each training set consists in very small 25 samples (total for both 2 classes) from static sampling window. For testing, 1024 data points uniformly sampled from the entire grid (32*32 points) are considered. The experiments contain a total of 400 steps. Different drift rates are applied on separated datasets :a constant drift rate of $2\pi/400=0.016$ rad/time step ; a exponentially increasing drift rate (the board rotates increasingly faster) ; a sinusoidal varying drift rate ; and a Gaussian pulse shaped (slow-fast-very fast-fast-slow) drift rate.. We have totally four separated datasets with four different drift rates (we call them CB Constant, CB Sinusoidal, CB exponential, CB pulse respectfully). For each one, we evaluate mOC-iSVM with the exact protocol proposed by Elwell [?] and we include supplementary three results with other size of incoming data packets (50, 100, 200 examples for each step).

For the implementation of mOC-iSVM.AP, we used LibSVM [?] in Matlab. The kernel function used is the classical RBF. In those experiments, all experiment on SEA and CB dataset are realized with $age = 1$ to maximize the dynamic property of system. During training, the ν parameter of SVM approximating the fraction of training errors over support vectors is fixed at 0.9 to maximize the specificity of the model. To simplify the interpretation of the analysis and save up time of execution, we fix the same parameters for all steps of each experiment without using the usual grid search selection approach like in [?]. Anyway, a grid search is always available to optimize the results, but with a quite high runtime cost.

4.2 Performances Evaluation For mOC-iSVM.AP

The figure 3 and 4 present the summary of performance of mOC-iSVM.AP on SEA and Checker Board

(CB) datasets. The comparisons are made with the ensemble of classifier methods as L++.NSE, Adaboost, SEA (Streaming Ensemble Algorithm), DWM (Dynamic weighted majority) in three different core of classifier : Naive Bayes (NB), SVM, and CART. The results with the single classifier of NB, SVM and CART are included too. We compare mOC-iSVM.AP with his ancestor mOC-iSVM and further, we include also the results of mOC-iSVM.AP in different size of data : packet 500, 1000 examples for SEA dataset ; packet 50, 100, 200 examples for CB dataset. The evaluation takes the average value of accuracy² of all steps as the principal measure with his standard deviation.

For the evaluation on SEA dataset, our method reduced performances than L++.NSE, DWM. It is similar to SEA, and outperforms Adaboost (NB, CART and SVM core). But with other size of incoming data (500 and 1000), we outperform all of them. For the evaluation in CB dataset, our method is almost higher than all other methods. In the special case of sinusoidal drift rate, we are much higher (more 5%) than the second place (L++.NSE). Globally, our mOC-iSVM.AP approach demonstrates a very good adaptation to the drifts with certain dependence in the size of incoming data. It proves its ability to adapt its recognition for non-stationary environment.

Compared to mOC-iSVM.AP, its ancestor version mOC-iSVM shows a poor accuracy (compared to L++.NSE in dynamic environment). In the CB dataset, it outperforms all classifier methods (as L++.NSE, Single, DWM, SEA, Adaboost) using NB core, it shows similar results than all ones using CART core, but less accurate than all methods using SVM core. One reason also proposed in [9] is that one-class can not expect as good results as binary SVM for classification purpose due to the absence of negative information in the training. Moreover a second reason that explains why mOC-iSVM has a weak accuracy in the non-stationary environments is due to its slow reaction to change (as explain in section 3). On the curve illustrated in figure 4, with a high stability setting and certainly a high specificity of model ($\nu = 0.9$), we can observe that mOC-iSVM obtains high accuracies in stable situations. But when the drift happens (step 50, 100 and 150), the old knowledge (represented by ancient SVs) continue to influence the class representation and renders the system more slow to react to any changes. Consequently, even though the accuracy increases on each step (from 50 to 100, 100 to 150 and 150 to 200), the speed is still very slow to acquire high results. On the other hand,

². Accuracy = (True Positive + True Negative)/(True Positive + True Negative + False Positive + False Negative)

Approach	Accuracy	Std
L++.NSE (NB)	96.6	0.2
Single (NB)	94.7	0.6
DWM (NB)	96.6	0.6
SEA (NB)	95.4	0.4
Adaboost (NB)	93.2	0.4
L++.NSE (SVM)	96.8	0.2
Single (SVM)	95.6	0.4
SEA (SVM)	95.7	0.2
Adaboost (SVM)	93.2	0.2
L++.NSE (CART)	95.8	0.5
Single (CART)	86.7	1
SEA (CART)	95.6	0.3
Adaboost (CART)	93.3	0.5
mOCISVM	93	4.3
mOCISVM-AP	95.1	2.2
mOCISVM-AP (Packet 500)	96.5	1.7
mOCISVM-AP (Packet 1000)	97.3	1.4

FIGURE 2 – Summary of SEA dataset evaluation

Approach (Accuracy %)	CB Constant	Std	CB Pulse	Std	CB Exp	Std	CB Sin	Std
L++.NSE (NB)	69.9	1.3	70.5	1.6	69.1	1.4	71.1	1.5
Single (NB)	56.6	1.7	54.3	1.7	56.5	1.7	55.3	1.7
DWM (NB)	59.6	1.6	56.4	1.7	59.6	1.7	57.9	1.7
SEA (NB)	60.1	1.2	64.2	1.6	61	1.3	63.7	1.4
Adaboost (NB)	59.9	1.7	59	1.8	59.9	1.6	59.2	1.7
L++.NSE (SVM)	81.9	0.9	84	0.7	81.6	0.9	83.5	0.9
Single (SVM)	76.6	1.5	79.9	1.5	76.5	1.5	78.6	1.5
SEA (SVM)	71.6	0.7	78.5	0.6	73	0.7	75.4	0.6
Adaboost (SVM)	81	1	82	1.1	80.2	1.1	81.9	1.1
L++.NSE (CART)	77.3	1.1	81.2	1	77	1.1	79.4	1
Single (CART)	67.8	1.9	69.3	2.6	67.7	1.9	68.7	2.3
SEA (CART)	69.3	0.9	77.2	0.8	70.6	0.9	73.2	0.8
Adaboost (CART)	74.7	1.3	80.5	1	74.8	1.3	77.4	1.2
mOCISVM	73.1	5.4	75.1	8.8	72	6.9	74	5.7
mOCISVM-AP	81.8	3.9	85	4.1	83.4	3	87	5.4
mOCISVM-AP (packet 50)	85	3.6	85.6	4.5	88.1	2	87.5	4.1
mOCISVM-AP (packet 100)	87	2.4	87	5.6	89.7	2	85.9	4.3
mOCISVM-AP (packet 200)	86.7	2.3	86.9	7.9	84.3	4	84.5	4.2

FIGURE 3 – Summary of CB dataset evaluation

thanks to this stability, when a similar situation appears (than encountered previously in the first steps), the system does not loss so much in accuracy, and can increase very quickly its performance. Another quality of the model has been encountered when the one-class is tuned with a high plasticity (with a high generality introduced by small values of ν , like 0.0005). The consequence is a good adaptation of the system to the environment (because of good ratio between old and new knowledge) but with a lowest accuracy due to the lack of precision of SVs to modelize the concepts boundaries.

One sensitive point in mOC-iSVM.AP is the time that is required to the system to be well adapted to environment changes, mainly caused by the parameters setting that influence the SVs selection. Five curves illustrating the mOC-iSVM.AP accuracy in different context are illustrated in figures 5, 6, 7, 8 and 9. We can notice that with a small packet of incoming new data, the effect of the choices for the selection of new support vectors (through SVM parameterization and old SVs consideration) can damage the accuracy because we effect directly on the boundary of the concept (the chosen support vectors) that will influence directly

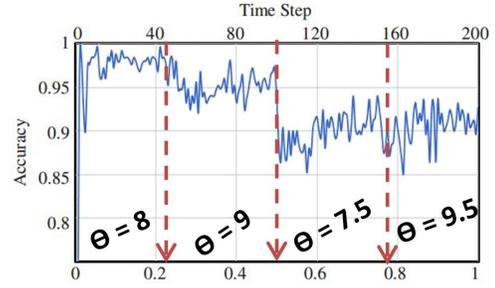


FIGURE 4 – mOC-iSVM on SEA Dataset with $\nu = 0.9$

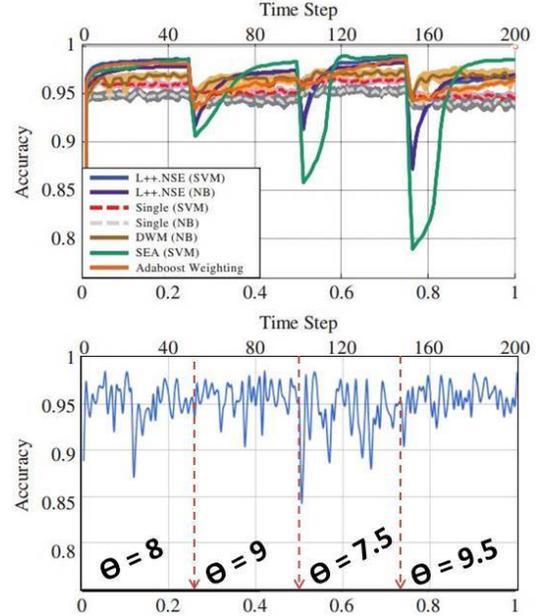


FIGURE 5 – SEA dataset with mOC-iSVM.AP and other methods

on the model. This phenomenon has been encountered in situations where the environment changes abruptly and is perceptible on local variations in the curves. But of course, the only consideration of a big packet of incoming new data is not enough to increase the system adaptation, because even with it, the choice of new SVs is still dependent to the distribution of the training dataset compared with the test dataset. Anyway, in order to save up more processing time, we have fixed all parameter without using the grid search approach (as explained in previous section). It has consequently increased the average standard deviations over the experiments (see figure 2 and 3).

Let's consider now mOC-iSVM.AP overall performances compared to other state-of-the-art approaches in non-stationary environments.

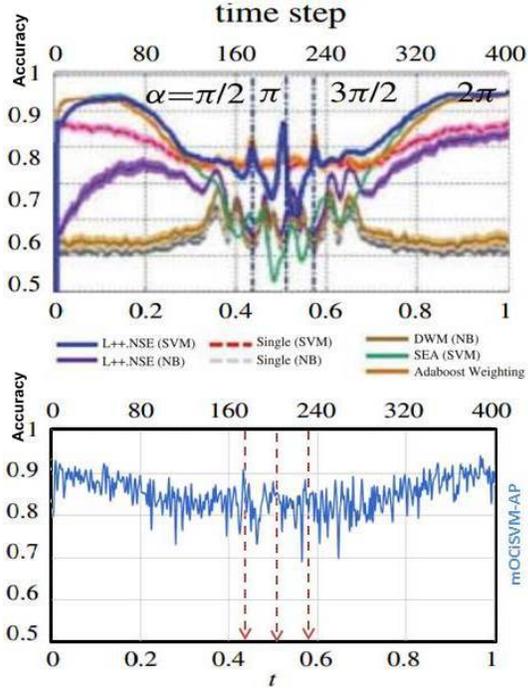


FIGURE 6 – CB Pulse dataset with mOC-iSVM.AP and other methods

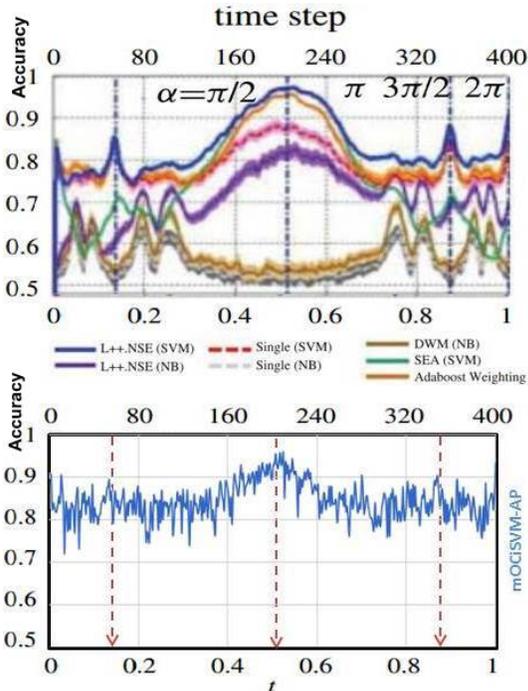


FIGURE 7 – CB Sinusoidal dataset with mOC-iSVM.AP and other methods

First of all, by considering curves amplitudes in figures 5, 6, 7, 8 and 9, we can observe that our results outperform all other approaches, except L++.NSE-SVM. Globally, our method shows a greater stability in its accuracy representation over the time than the high performant L++.NSE-SVM approach without any drop of accuracy, especially in figure 6 and 7. The consideration of the more recent SVs ($age = 1$) in mOC-iSVM.AP makes the local adaptation easier when environment change abruptly but as a consequence, it also limits its ability to improve its accuracy over stable environment or if new incoming data are similar to ancient (the recurring environment). One necessary adaptation here is to offer to users the possibility to decide which is the best adapted age for preserving a high accuracy of the system.

In L++.NSE approach the class boundaries depend on the vote of many classifiers. Consequently changes append only when most of classifiers recognize them. Other methods based on set of classifiers increase their capacity to change the class boundaries by considering huge quantity of data. They get a good stability of decision compared to us, which is observable in their stable curves and low standard deviations over steps. But for those methods, this stability leads also to a difficulty to adapt efficiently to sharp drifts, like it is encountered between steps 180 to 220 on Gaussian pulse drift rate (figure 6), and steps 0 to 60 and 340 to 400 on sinusoidal drift rate (figure 7).

5 Experiments In Stationary Environment

L++.NSE has been designed for non-stationary environment using Learn++ approach. The version for stable environment of Learn++ approach is L++.SVM and L++.NC. Their performances in stationary environment have been compared to our initial proposition mOC-iSVM in [?]. In stationary environment, an advantage of mOC-iSVM.AP is its ability to adapt very easily to its environment by just changing the chosen age of system. In this experimental part, we intend to show that our new proposal is more performant than mOC-iSVM in stationary environment by using efficiently old knowledge. So we design a scenario based on images dataset that illustrates the removal of a concept in some steps and its reappearance after. The system is tuned to recall the old knowledge and improve the resulting accuracy.

The database that is used for the experiments is a real problem from the Digidoc ANR Project (Histo-

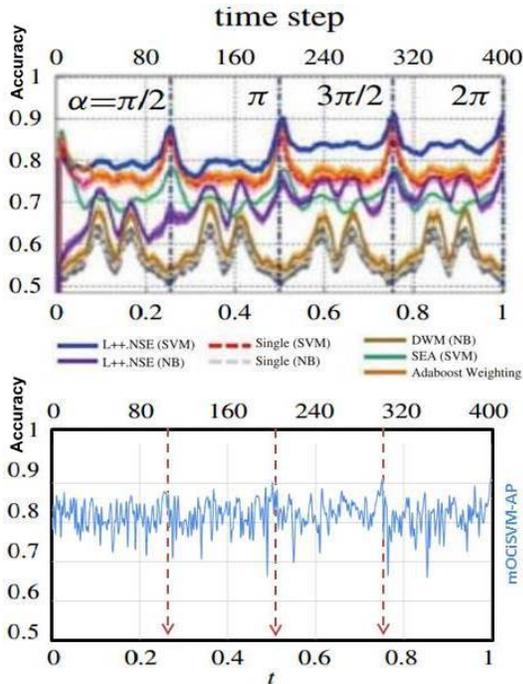


FIGURE 8 – CB Constant dataset with mOC-iSVM.AP and other methods

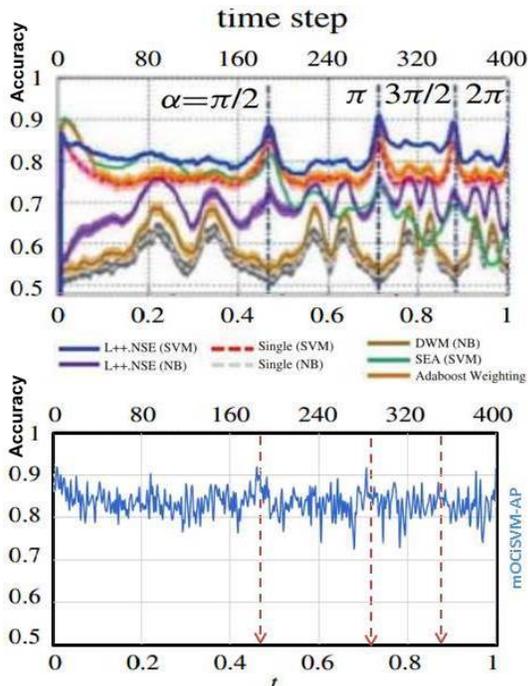


FIGURE 9 – CB Exponential dataset with mOC-iSVM.AP and other methods

rical Document Digitization on data flow process). It contains six classes of document images, each one with a different contents : Handwritten musical scores (490 instances), Printing I (230 instances), Printing II (99 instances), Handwriting I (379 instances), Handwriting II (103 instances), Maps (279 instances). The classes Printing I and Printing II are composed of two categories of printed styles : we consider them as two subclasses of an unique more general class of Printing. The same principle has been applied for Handwriting I and II. The documents are initially represented with a set of 41 features that have been used for their ability to characterize the variety of contents. Then, we have selected 26 features among the 41 thanks to the SVM-RFE method [?]. The documents of the dataset present visual evident differences of contents. So it is helpful to adapt the description to the content of each class (printed, handwritten, strokes bases, drawings documents).

During training, the ν parameter of SVM is fixed at 0.01. The γ parameter for the Gaussian kernel selected by a grid search varies between 0.05 and 1. Those values have given the best accuracy on preliminary tests. So as to improve the adaptation (maximize the stability property of system) to the stationary environment, the maximum age is chosen in the system : all SVs from the youngest to the oldest ones are used for training. The main performance metric used is the Average of Balanced Accuracy [?] over all classes (Balanced Accuracy = (Recall + Specificity)/2). This expression of accuracy is a good compromise between the detection of true positives and the rejection of true negatives, and allows taking into account the effect of the imbalanced dataset.

Firstly, the system is trained (steps 1-6) on two kinds of subclasses (Handwriting I,II and Printing I,II) respectively labeled as Handwriting and Printing . Next, from step 7 to 10, examples from one of two subclasses (Handwriting II and Printing II) are also coming alone with the single labels (Handwriting and Printing) while others subclasses data (for classes Handwriting I and Printing I) disappear in these steps. Doing so, the concepts of Handwriting (resp. Printing) lost a part of their description concerning Handwriting I (reps. Printing I). Finally, from step 11 to 12, the training dataset comes back to the initial situation with the consideration of the two initial kinds of subclasses (Handwriting I and Printing I). Of course, the test set used is independent to these modifications.

Figure 10 shows the performance of mOC-iSVM.AP compared to mOC-iSVM in the situation of stationary environment . As display in the curve, in the first steps of learning (and recognition), the results of two me-

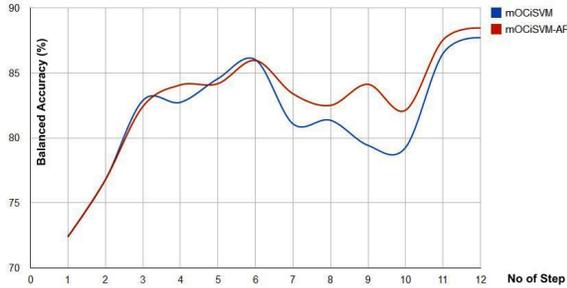


FIGURE 10 – mOC-iSVM and mOC-iSVM.AP on Digidoc Dataset

Runtime(s)	SEA Dataset	CB Dataset
L++.NSE	50.32	71.35
SEA	10.41	12.97
DWM	21.11	33.17
Adaboost	82.88	115.24
mOCiSVM.AP	67.6	84.17
mOCiSVM	76.61	90.11

FIGURE 11 – Summary of Runtime in Second (S)

thods are almost the same. When the classes Handwriting and Printing appear alone (without their respective subclasses) the mOC-iSVM.AP (red curve) outperforms mOC-iSVM (blue curve). Precisely in the 9th step, the difference between the performance of the ancient and new version of mOC-iSVM is positive over 5% increased accuracy.

6 Runtime

The runtime of mOC-iSVM and mOC-iSVM.AP depends on the number of data in the training set and also the research space of parameters. The ν parameter decides how many support vectors must be included in the system in relation to the number of training data. The runtime is increased when the ν increases and it increases more when the research space is larger. The values indicated in the runtime summary (figure 11) correspond to the above experiments.

In the runtime summary below (unit of time : seconde), our approach is less time consuming than the ancient version. It is a positive point of our new mOC-iSVM.AP proposition. Our results are closer L++.NSE but still much lower than DWM and SEA.

7 Conclusion

In this paper, we have shown the efficiency of a new incremental mechanism called mOC-iSVM.AP to adapt the recognition in stationary and non stationary

environment. Thanks to the use of age hierarchy of support vectors and the independent modeling of the concepts, the classifier can, in the dynamic environment, easily learn, and quickly adapt its context to sudden modifications. In stable environment, it has a great ability to conserve important knowledge.

The learning system can be improved by taking benefits of rejection options of the one-class system. An improved user interaction can also be introduced to help the adaptation by guiding the age selection.

8 Acknowledgment

This work was supported by the ANR (French National Agency for Research) within the DIGIDOC project (ANR-10-CORD-0020).

Références

- [AESM10] A.Almaksour, E.Anquetil, S.Quiniou, and M.Chériet. Evolving fuzzy classifiers application to incremental learning of handwritten gesture recognition. *ICPR2010, Istanbul, Turkey*, 2010.
- [AKHNJY⁺13] Anh-Khoi.Ngo-Ho, N.Ragot, J-Y.Ramel, V.Eglin, and N.Sidère. Document classification in a non-stationary environment : a one class svm approach. *ICDAR'13, Washington DC, USA*, 2013.
- [CC11] C.C.Chang and C.L.Lin. Libsvm : A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2011.
- [CCC03] C.Hsu, C.Chang, and C.Lin. A practical guide to support vector classification. *LibSVM Guide, Version 2003*, 2003.
- [CM07] C.Alippi and M.Roveri. Just-in-time adaptive classifiers in non-stationary conditions. *IJCNN'07, New York, USA*, 2007.
- [CP01] G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. *NIPS'01, vol. 13*, 2001.
- [GM93] G.Widmer and M.Kubat. Effective learning in dynamic environments by

- explicit context tracking. *ECML'93, Berlin, Germany*, 1993.
- [GSN+92] G.A.Carpenter, S.Grossberg, N.Markuzon, J.H.Reynolds, and D.B.Rosen. Fuzzy artmap a neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Trans. Neural Netw, vol. 3*, 1992.
- [HWPJ03] H.Wang, W.Fan, P.S.Yu, and J.Han. Mining concept-drifting data streams classifiers. *ACM SIGKDD'03, New York, USA*, 2003.
- [IJSV02] I.Guyon, J.Weston, S.Barnhill, and V.Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning, Volume 46, Issue 1-3, pp 389-422*, 2002.
- [J.S93] J.S.R.Jang. Anfis : adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics, IEEE Transactions on May/June*, 1993.
- [KCKJ01] K.H.Brodersen, C.S.Ong, K.E.Stephan, and J.M.Buhmann. The balanced accuracy and its posterior distribution. *ICPR'10, Istanbul, Turkey*, 2001.
- [LFB01] L.Ralaivola and F.d'Alché-Buc. Incremental support vector machine learning : a local approach. *In Proc. of ICANN'01*, 2001.
- [LM01] L.M.Manevitz and M.Yousef. One-class svms for document classification. *Journal of Mach. Learn. Res. 2 : 139-154*, 2001.
- [MAA12] M.Bouillon, É.Anquetil, and A.Almaksour. Decremental learning of evolving fuzzy inference systems using a sliding window. *ICMLA'12, Florida, USA*, 2012.
- [MI10] M.Karasuyama and I.Takeuchi. Multiple incremental decremental learning of support vector machines. *IEEE Trans. on Neural Networks, Vol. 21 Issue 7, pp. 1048-1059*, 2010.
- [MRP10] M.N.Kapp, R.Sabourin, and P.Maupin. Adaptive incremental learning with an ensemble of support vector machines. *ICPR'10, Page(s) : 4048- 4051*, 2010.
- [N.C01] N.C.Oza. Online ensemble learning. *PhD thesis, University of California, Berkeley, USA*, 2001.
- [NHK99] N.Syed, H.Liu, , and K.Sung. Incremental learning with support vector machines. *IJCAI'99, Stockholm, Sweden*, 1999.
- [NM94] N.Littlestone and M.K.Warmuth. The weighted majority algorithm (wma). *Inform. Computation, 108 :212 261*, 1994.
- [PCSM01] P.Laskov, C.Gehl, S.Kruger, and K.-R. Muller. Incremental support vector learning : Analysis, implementation and applications. *Journal of Machine Learning Research, vol. 7, pp. 1909-1936*, 2001.
- [RLSV01] R.Polikar, L.Udpa, S.Udpa, and V.Honavar. An incremental learning algorithm for supervised neural networks. *IEEE Trans.on SMC, Special Issue on Knowledge Management*, 2001.
- [RR11] R.Elwell and R.Polikar. Incremental learning of concept drifts in non-stationary environments. *IEEE Transactions on Neural Networks*, 2011.
- [SPT05] A. Shilton, M. Palaniswami, and D. Ralph Ah Chung Tsoi. Incremental training of support vector machines. *IEEE Trans. on Neural Networks, Vol. 16 Issue 1, pp. 114-131*, 2005.
- [S.R01] S.Ruping. Incremental learning with support vector machines. *ICDM01, San Jose, California, USA*, 2001.
- [WY01] W.N.Street and Y.S.Kim. A streaming ensemble algorithm (sea) for large-scale classification. *ACM SIGKDD'01*, 2001.
- [YA05] Y.Prudent and A.Ennaji. A new learning algorithm for incremental self-organizing maps. *ESANN'05, Bruges, Belgium*, 2005.