

---

# Interpretable Cascade Classifiers with Abstention

---

**Matthieu Clertant**  
NutriOmics  
INSERM  
Sorbonne University  
Paris, France

**Nataliya Sokolovska**  
NutriOmics  
INSERM  
Sorbonne University  
Paris, France

**Yann Chevaleyre**  
LAMSADE  
Dauphine University  
PSL Research University  
UMR CNRS 7243  
Paris, France

**Blaise Hanczar**  
IBISC  
Paris Saclay University  
Evry University  
Evry, France

## Abstract

In many prediction tasks such as medical diagnostics, sequential decisions are crucial to provide optimal individual treatment. Budget in real-life applications is always limited, and it can represent any limited resource such as time, money, or side effects of medications. In this contribution, we develop a POMDP-based framework to learn cost-sensitive heterogeneous cascading systems. We provide both the theoretical support for the introduced approach and the intuition behind it. We evaluate our novel method on some standard benchmarks, and we discuss how the learned models can be interpreted by human experts.

## 1 Introduction

In medical applications, there is an acute need for sequential rules to acquire patients data, since medical measurements vary tremendously in acquisition cost and in predictive power. There exist numerous examples from clinical practice that illustrate the importance to find a trade off between predictive error of a model and the cost of features used by this model. E.g., the type 2 diabetes remission after a bariatric surgery can be predicted using a set of cheap clinical parameters such as diabetes duration, age, etc., and the prediction is quite accurate for some subjects [1]. However, for a particular group of individuals such a simple model is not accurate enough, and to increase the predictive accuracy, it is advised to measure the C-

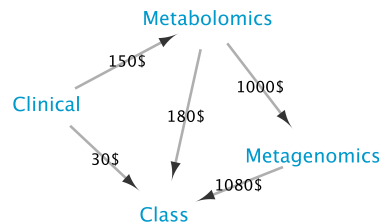


Figure 1: An example of an individualized diagnostic protocol: heterogeneous data sources involved in the health status prediction, and their corresponding costs.

peptide what is a rather expensive and invasive procedure, and should be avoided for majority of individuals [11]. In this case the clinicians would prefer a cascade classifier which rejects a decision at the first stage if it is not accurate enough, and if there is a hope that a more expensive additional measurement would provide a more accurate patients stratification. Another motivating example is illustrated by Figure 1 that shows that a patient can be stratified using either cheap clinical parameters (30 USD), or clinical and metabolomics data (what is more expensive and costs 180 USD), or that the expensive metagenomics measurements are needed in addition to the clinical and metabolomics data (with the overall cost more than 1000 USD).

*Dynamic diagnostic protocols* are individually tailored acquisitions of patients data with the aim to provide the most accurate diagnostics for the lowest cost. In contrast with standard approach where all patients follow the same medical protocol, dynamic patients treatment incorporates heterogeneous data, and also the order of medical analyses can vary from one patient to another. Such a dynamic process of patients data acquisition can be called an adaptive strategy.

The goal of cascade classifiers under budget constraints

is to classify examples with low cost, and to minimise the number of expensive or time-consuming features (or measurements). Here, under cost we mean the cost paid by a learner during training to acquire relevant information, and at performance time, there is a cost to pay to get relevant information about the current observation. The cost can reflect money or time for data acquisition, or even side effects of a treatment.

We adopt a POMDP (Partially Observable Markov Decision Process) framework to model individual optimal policies under budget constraints. In contrast to much of the existing literature, *we propose a method which simultaneously learns classification and rejection models together with the personalized feature selection.*

Our contribution is multi-fold, and can be summarized as follows:

- We propose a POMDP framework for handling a general dynamic diagnostic protocol setting, and introduce an original methodology to learn individual adaptive strategies;
- Our method learns cost-sensitive heterogeneous cascading systems;
- We show how a model learned with the proposed method can be interpreted, and how its output can be explained by human experts;
- Numerical experiments on benchmark data show that the proposed method achieves the state-of-the-art performance in terms of accuracy. The qualitative evaluation illustrates that the individualised dynamic strategies allow to stratify observations what can be used in the methods of personalised medicine.

The paper is organised as follows. We discuss the related work in Section 2. We introduce our approach in Section 3. The interpretable deep learning methods are discussed in Section 4. Numerical experiments are shown in Section 5. Concluding remarks and perspectives close the paper.

## 2 Related Work

In this work, we focus on multi-stage sequential reject classifiers that reduce the cost of data acquisition. A number of methods to post-process classifiers in order to reduce their test time complexity were proposed. The largest part of them uses a cascade of classifiers with a reject option. In a cascade, at each stage, a classifier can either classify an input or reject it, and send it to the next classifiers [22, 3]. Most cascade classifiers are usually designed for binary problems, and

aim to reduce computational cost during prediction. The multi-stage cascade classifiers deal with multi-class problems, and they can make a classification decision at any stage.

The learning with rejection framework aims to learn simultaneously two functions: a classifier to label an observation and a rejection function [6]. The trade-off between predictive accuracy and rejection rate was studied by [4] where the reject classifiers were considered in the Bayesian setting.

Recent works in the non-Bayesian scenario introduce classifiers with a reject option where the reject region is defined via a distance to the separating hyperplane. A seminal work of [2] provided a theoretical analysis for a discontinuous loss function taking into account the rejection cost.

The problem of classification with reject where a classifier has an option to abstain from taking decision about a label, was discussed in a number of recent publications, and there have been several attempts to integrate a reject option in the state-of-the-art classifiers such as support vector machines [9, 7]. A cascade system of classifiers with a reject option which minimizes the cost to stratify patients was introduced by [8]. The idea to construct a cascade classifier using trees was exploited by several research teams, e.g., [12] formalised the problem of learning cost-effective models as a Markov decision tree, and employed a variant of the upper confidence bound for trees (UCT). A two-stage algorithm based on random forests and an efficient pruning of all trees simultaneously was proposed by [16].

The most important aspect of cascade classifiers is to learn a function which decides whether to reject an observation or to label it. A problem how to learn a function which is able to identify regions where a low prediction cost model is sufficient compared to a high prediction cost model was considered by [15]. At test time, the method uses the gating function to choose a prediction model.

In [21], the problem of supervised sequential classification under budget is formulated as a Markov Decision Process (MDP). However, the order of measurements or features is fixed and the paper focuses to find reject regions based on known decision boundaries, which means that the classification models of all stages are learned in advance and are available. The cascade problem is formulated as a simple dynamic programming, and the problem reduced to learning reject regions which are subject specific.

Reinforcement learning is actively explored to prescribe optimal treatment where an assessment score

of patients status is used as a reward. A combination of hidden Markov models and deep Q-networks was shown to learn optimal individual heparin dosing for intensive care unit patients [17]. Another combination of modern machine learning methods, namely, of a sparse autoencoder and policy iteration was used in [24] to model personalized optimal glycemic trajectories. [18] has shown that optimal quantity of intravenous medication can be estimated with double deep Q-learning. Tabular Q-learning was proposed in [20] to recommend treatment for schizophrenia patients. A medication strategy for non-small cell lung cancer can be learned by approximating the Q-function using a support vector regression which can incorporate censored data [25].

Recently [23] pointed out that the modern dynamic treatment recommendation systems are learned either by a supervised method, or using reinforcement learning. To combine their benefits, [23] introduced a supervised reinforcement approach where recurrent neural networks are used to solve a POMDP.

### 3 From POMDP to MDP in Cascade Learning with Reject Option

#### 3.1 Context and Preliminaries

In real-world decision making applications, uncertainty comes from two sources: uncertainty related to the effects of actions and uncertainty about the current state, i.e., partial observability. The POMDP is a widely used framework to deal with these uncertainties. Compared to fully observable MDPs, the uncertainty about the current state is a characteristic of POMDPs. In our scenario, the features are gradually observed along the cascade, and a decision can be made from partial observations. The problem can naturally be modeled as a POMDP.

In the problem of diagnostic protocol learning, at each stage of a cascade classifier, the system decides whether to classify an observation, or to continue data acquisition. The process of data acquisition ends when the system is confident enough to return a prediction. What is specific to the task of diagnostic protocol learning, it is that taking an action, i.e., new data acquisition, does not change the state of the patient. Such a POMDP is known as a purely epistemic MDP. A *purely epistemic MDP* [19] is a much less studied framework, where the actions bring new information without changing the current state.

It is challenging to solve a POMDP, and the problem is often reformulated as a MDP with extended states, e.g., the *belief-state MDP* or the *history-state MDP* (see e.g. [10]). For the sake of simplicity, we choose to

build a history-state MDP, where each state consists of the history of all measurements done so far on a patient. Note that such a state does not contain unobserved features or the class of interest. This leads to a fully observable MDP, where standard reinforcement learning algorithms can be applied.

To facilitate the reading, we provide the most important notations in Table 1. Here, for a purpose of clar-

Table 1: Important notations

Notation	Description
$L$	Number of classes
$K$	Number of features
$t$	Current depth in the cascade
$[N]$	Set of integers from 1 to $N$
$(x^i, y^i)_{i \in [N]}$	Observations with classes
$x_k$	Feature value $k$
$x_{[t]}$	Feature values collected prior to $t$
$a_t$	Action at step $t$ , < 0 : feature index, > 0 : class label
$a_{[t]}$	Vector of actions prior to $t$
$c$	Cost function
$V^\Pi(s)$	Expected cost from state $s$ under policy $\Pi$
$Q^\Pi(s, a)$	Cost-to-go to $a$ under policy $\Pi$
$Q_w(s, a)$	Approximation of function $Q$
$f_w$	Classifier
$r_w$	Rejection/selection model
$M_{[t]}$	Mask corresponding to $a_{[t]}$
$\beta_w(\cdot)$	Weight function of softmax regression

ity, we present our methods with two restrictions. In real life applications, (1) a measurement may be associated with a group of features as a matrix, sequence, graph or hyper-graph and (2) the cost of a wrong classification depends on the estimated label (e.g false positive or false negative). To simplify the presentation but without any loss of generality, we consider that each measurement is related to one feature only. All the results and observations presented below can be extended directly to any shape of data. We also associate a cost of 1 to any misclassification. Differentiated weights are hyperparameters of the cascade classifier which can be integrated directly.

#### 3.2 Problem Statement

Let  $(x^i, y^i)_{i \in [N]}$  be a training sample from a distribution  $D : (X, Y) \sim D$ . We will omit the superscript  $i$  in the definition of the supervised learning problem: predicting  $y$  from partial or fully observable  $x$ . The dependent variable  $y$  belongs to the set

of classes  $[L] = \{1, \dots, L\}$ . The variable  $x$  comes from a large set  $\mathcal{X}$  consisting of  $K$  subspaces of features:  $x = (x_k)_{k \in [K]} \in \mathcal{X}$  with  $\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_K$ . Each feature  $k$  corresponds to a specific measurement associated to an acquisition cost  $\delta_k$ . At each step of the cascade, the decision process can either choose to observe a new feature or to stop and to return the class  $\hat{y}$ . The cost of a wrong classification is 1. The aim is to determine a policy  $\pi$  of actions minimizing the global cost.

At step  $t \in [K]$ , the action  $a_t$  recommended by the policy belongs to  $\{-K, \dots, L\}$ . In particular, we define two types of actions:

1.  $a_t \in \{-K, \dots, -1\}$ , the policy recommends to acquire feature of index  $-a_t$ .
2.  $a_t \in \{1, \dots, L\}$ , the policy recommends to stop acquiring new features, and to provided the predicted label  $a_t$ .

A third (degenerate) category corresponds to the case where the process has been stopped at a previous step:  $a_t = 0$  if and only if  $a_{t-1} \geq 0$ .

### 3.3 Cascade with Abstention

The state  $s_t$  summarizes all the observations from step 1 to  $t$ :  $s_t = (a_{[t]}; x_{[t]})$ , where  $a_{[t]} = (a_i)_{i \in [t]}$  and  $x_{[t]} = (x_{-a_i})_{i \in [t]}$  with the convention  $x_{-a_i} = \emptyset$  when  $a_i \geq 0$ . For the sake of compactness in the following definitions, we set  $s_0 = \emptyset$ . The cost function is:

$$c(s_t) = \begin{cases} \mathbb{1}_{a_t \neq y} \text{ with } y \sim Y \mid s_t & , \text{ if } a_t > 0, \\ \delta_{-a_t} & , \text{ if } a_t < 0, \\ 0 & , \text{ otherwise.} \end{cases} \quad (1)$$

If  $a_t = 0$ , the cost is null (third circumstance). Note that the cost at step  $t$  strongly depends on the last action  $a_t$  which does not appear explicitly in the parameter as it is included in  $s_t$ . The global cost  $C$  and the cost from step  $k$ , noted  $C_k$ , are given by:

$$C(s_K) = \sum_{t \in [K]} c(s_t) \text{ and } C_t(s_K) = \sum_{k \in [t:K]} c(s_k), \quad (2)$$

where  $[t:K] = \{t, \dots, K\}$ . The policy  $\pi$  is deterministic and maps the action  $a_t$  with the available data  $s_{t-1}$ :  $a_t = \pi(s_{t-1})$ . Let  $C^\pi$  be the expected cost associated to policy  $\pi$ :  $C^\pi = \mathbb{E}_D[C(s_K)]$ . The cost-to-go from step  $t$ , noted  $V^\pi$ , only depends on policy and current state  $s_{t-1}$ :

$$V^\pi(s_{t-1}) = \mathbb{E}_D[C_t(s_K) \mid s_{t-1}]. \quad (3)$$

$Q^\pi(s_{t-1}, a_t)$  denotes the cost to take a further step in the direction  $a_t$ , also named the cost-to-go to  $a_t$ :

$$Q^\pi(s_{t-1}, a_t) = \mathbb{E}_D[C_t(s_K) \mid s_{t-1}, a_t]. \quad (4)$$

According to the nature of the action, we have:

$$Q^\pi(s_{t-1}, a_t) = \begin{cases} \mathbb{P}_D(a_t \neq Y \mid s_{t-1}) & , \text{ if } a_t > 0 \\ \delta_{a_t} + V^\pi(s_t) & , \text{ if } a_t < 0, \\ 0 & , \text{ if } a_t = 0. \end{cases}$$

We emphasize that the order of actions prior to step  $t$  does not modify the cost-to-go from step  $t$ . The accuracy of the model based on  $s_{t-1}$  is compared to the acquisition cost of new features and the potential gain in terms of accuracy. Thus, the decision at step  $t$  is based on the current state  $s_{t-1}$ , the ‘‘historic state’’ of observations to step  $t$ , but their order does not play any role. The vector  $a_{[t-1]}$  might then be thought as a vector of indices to identify the feature values  $x_{[t-1]}$ .

The following properties give some insights in the optimal solution  $\pi^*$ :

$$\pi^* = \arg \min_{\pi} C^\pi. \quad (5)$$

**Theorem 1.** *Let  $\hat{y}$  and  $\hat{r}$  be two natural candidates in case of classification and rejection when the distribution  $D$  is known:*

$$\hat{y} = \arg \max_{a_t \in [L]} \mathbb{P}_D(a_t = Y \mid s_{t-1}), \quad (6)$$

and

$$\hat{r} = - \arg \min_{a_t \in [-K:-1]} Q^\pi(s_{t-1}, a_t). \quad (7)$$

For all  $t \in [K]$  (or to the end of the cascade), the optimal solution of the MDP system satisfies:

$$\pi^*(s_{t-1}) = a_t^* = \begin{cases} \hat{y} & , \text{ if } \max_{a_t > 0} \mathbb{P}_D(a_t = Y \mid s_{t-1}) \\ > 1 - \min_{a_t < 0} Q^\pi(s_{t-1}, a_t) \\ \hat{r} & , \text{ otherwise.} \end{cases} \quad (8)$$

*Proof.* The trivial case where the cascade has been stopped at a previous step is not considered in this proof. Finding the optimal solution reduces to optimize each single stage, and at step  $t$ , we have:

$$a_t^* = \arg \min_{a_t \in [-K:L] \setminus \{0\}} Q^\Pi(s_{t-1}, a_t). \quad (9)$$

From there,

$$\begin{aligned} a_t^* &= \arg \min_{a_t \in [-K:L] \setminus \{0\}} [\mathbb{P}_D(a_t \neq Y \mid s_{t-1}) \mathbb{1}_{\{a_t > 0\}} \\ &\quad + Q^\Pi(s_{t-1}, a_t) \mathbb{1}_{\{a_t < 0\}}] \\ &= \arg \min_{a_t \in [-K:L] \setminus \{0\}} [\{1 - \mathbb{P}_D(a_t = Y \mid s_{t-1})\} \mathbb{1}_{\{a_t > 0\}} \\ &\quad + Q^\Pi(s_{t-1}, a_t) \mathbb{1}_{\{a_t < 0\}}]. \end{aligned}$$

Thus, the optimal decision is to stop the feature acquisition, and to recommend the class  $\hat{y}$  iff:

$$\begin{aligned} \min_{a_t \in [L]} [1 - \mathbb{P}_D(a_t = Y \mid s_{t-1})] \\ &< \min_{a_t \in [-K:-1]} Q^\Pi(s_{t-1}, a_t), \\ \max_{a_t \in [L]} \mathbb{P}_D(a_t = Y \mid s_{t-1}) \\ &> 1 - \min_{a_t \in [-K:-1]} Q^\Pi(s_{t-1}, a_t). \end{aligned}$$

If this inequality is not true, the optimal solution would be reject ( $\hat{r}$ ) in favor of a new measurement.  $\square$

## 4 Deep Models

In this section, we describe how we learn our cascading model. We apply the deep Q-learning to estimate it. We are interested in particular in an interpretable model which is able to stratify observations in several groups to promote development of methods of personalized medicine.

### 4.1 Deep Q-Learning

DQN (deep Q-network) [14] combines reinforcement learning with deep neural networks. We use the version with the experience replay that aims to randomize the data order and to remove correlations in the observation sequence. The complex function  $Q$  is approximated by one (or some) neural network(s) with weights  $w$ , and the training procedure consists to adjust the  $w$  to the data through the Bellman equation. At iteration  $j$ , the Q-learning process is based on the following objective function:

$$\mathcal{L}(w_j) = \mathbb{E}_{U(D)} \left[ L(y; Q_{w_j}(s, a)) \right], \quad (10)$$

with  $y = c + \gamma \min_{a'} Q_{w_j^-}(s', a')$ , where  $U(D)$  means that the sample  $(s, a, c, s')$  is drawn uniformly and where  $L$  denotes a loss function. In our problem formulation,  $\gamma = 1$ , since we cannot neglect any cost paid in the cascade. The parameters  $w_j$  are the parameters of the Q-network at iteration  $j$ , and  $w_j^-$  are the weights to compute the target at iteration  $j - 1$ .

The function  $Q$  is approximated by two deep neural networks,  $f_w$  and  $r_w$ , corresponding to the classifier ( $a > 0$ ) and the rejection/selection model ( $a < 0$ ).

$$Q^\pi(s, a) \approx Q_w(s, a) = \begin{cases} f_w(s, a) & , \text{ if } a > 0 \\ r_w(s, a) & , \text{ if } a < 0. \end{cases} \quad (11)$$

The policy estimate is obtained by using the result expressed in equation (8) (see Theorem 1, Section 3).

$$\Pi_w(s, a) = \begin{cases} y_w & , \text{ if } \min_{a>0} f_w(s, a) < \min_{a<0} r_w(s, a) \\ r_w & , \text{ otherwise,} \end{cases} \quad (12)$$

with  $y_w = \arg \min_{a>0} f_w(s, a)$  and  $r_w = \arg \min_{a<0} r_w(s, a)$ .

### 4.2 Interpretable Aspects of the Classifier

Although real-valued ‘‘black box’’ classifiers are powerful tools in medical diagnostics and prognostics, a model that is able to explain its predictions is of a big interest for clinicians [1]. We are aware that it is possible that there is a price to pay in accuracy for a model which is simple and easily interpretable.

The inputs of both neural networks ( $f_w$  and  $r_w$ ) are the current state:  $s_t = (a_{[t]}; x_{[t]})$ . The indices of features already observed at current step  $t$ , noted  $a_{[t]}$ , are encoded as a binary vector mask  $M_{[t]}$  of size  $K$ : 1 if the feature  $k$  has been observed, 0 otherwise. The value of the observed features  $x_{[t]}$  might be extracted from the vector  $x$  (all the features) by applying the element-wise product with  $M_{[t]}$ . Thus, the state  $s_t$  is summarized by  $(M_{[t]}, M_{[t]} \odot x)$ . In this vector of size  $2 \times K$ , an unobserved feature  $k$  is associated to a null value for the  $k$ -th elements of both  $M_{[t]}$  and  $M_{[t]} \odot x$ .

The classifier  $f_w$  is a deep neural network, composed of fully connected layers, and the last layer is based on a softmax regression. The fully connected layers provide the parameters  $\beta_w$  used in the final regression. They return a  $L \times K$  matrix of weights depending only on the mask:

$$\beta_w : (M_{[t]}) \rightarrow \beta_w(M_{[t]}) = (\beta_w^a(M_{[t]}))_{a \in [L]}, \quad (13)$$

where  $\beta_w^a(M_{[t]})$  is a vector of size  $K$ . The final layer corresponds to a softmax function taking the vector  $M_{[t]} \odot x$ :

$$f_w(s_t, a) = \frac{\exp \langle \beta_w^a(M_{[t]}), M_{[t]} \odot x \rangle}{\sum_{a>0} \exp \langle \beta_w^a(M_{[t]}), M_{[t]} \odot x \rangle}. \quad (14)$$

Thus, the values of the observed features,  $M_{[t]} \odot x$ , are used only in the last layer. This together with the small number of features used in most of the circumstances (see Figures 3 and 5) support the post hoc interpretability of the classifier. Moreover, if the human experts focus on feature  $k$  observed at time  $t$  in the cascade, the product  $[\beta_w^{a',k}(M_{[t]}) - \beta_w^{a,k}(M_{[t]})]x_k$  is the weight associated to the features of index  $k$  in the logarithm of the odds of action  $a$  against  $a'$ . This allows to assess easily the importance and the uncertainty associated to this feature in regard of the decision at time  $t$ .

As the interpretability based on the sparsity and the linearity could compromise the accuracy of our method, we choose to associate the classifier with a flexible rejection/selection model,  $r_w$  which is a standard deep neural network composed of fully connected

layers. Its ability to identify complex patterns yielding to rejection should allow our model to identify the complex cases for which extra information are needed. The input of the rejection/selection model is the vector  $(M_{[t]}, M_{[t]} \odot x)$  and its output are the approximated cost-to-go for all the actions which recommend the selection of a new features  $(Q_w(a, s))_{a \in \{-K, \dots, -1\}}$ .

### 4.3 Pretraining

There are two phases:

1. At first, the classifier is pretrained on all training data. Each observation is associated to a mask drawn randomly (uniform distribution).
2. The rejection/selection model can be pretrained jointly with the classifier by a run of Q-learning. This second phase is optional (details below).

In the second phase, the classifier has already been pretrained. We then focus on the rejection/selection model  $r_w$ . Here, the difference with the training relies on the input of  $r_w$ : only the masks are provided and not the values of features already observed. The input is  $(M_{[t]}, 0_K)$  with  $0_K$  a null vector of size  $K$ . An important consequence is that the rejection/selection model is not subject-specific anymore. For a certain mask (the labels of features already observed), the evaluated cost-to-go to a new feature,  $r_w(s, a)$  with  $a < 0$ , do not depends of  $x_{[t]}$ . The specific choice of rejecting or not is then based on the evaluated probability of each label,  $f_w(s, a)$  with  $a > 0$ . Thus, the second phase of pretraining corresponds to the training of a full cascade classifier model. However, the goal of this model is to identify a unique order for the cascade, optimised for all the subjects.

The model that we propose in this contribution is much more richer, since it is able to identify personalized order of features. During the final training phase, the unique order of exploration estimated during the pretraining is used as benchmark to evaluate paths which are now subject specific. The second and optional pretraining phase has two effects:

- It decreases the variability of the exploration path.
- It increases the accuracy and as a tradeoff, the number of features explored.

In the next section, we test the performance of two versions of the introduced interpretable cascade classifier with abstention. ICCA stands for our model with only the first phase of pretraining (classifier). When the two phases of pretraining are completed, the model is denoted p-ICCA.

## 5 Experiments

In this section, we illustrate the performance of the proposed approach. The existing cascading classifiers are mostly focused on learning rejection function [21, 16, 15] assuming that the classifiers are available. On the other hand, the literature on classifiers with rejection option [6, 5, 8, 7] studies the rejection function only. Therefore, although quite a rich literature on cascades and on rejections exist, to our knowledge, there is no any approach which is directly comparable to our dynamic feature selection method.

After some reflections, we decided to compare our approach to the random forests classifier. The model of random forest with feature selection can be viewed as a static version of a cascade classifier. Indeed, in our approach, the possible costs map with an average number of features observed. To make it comparable, the random forest is based on the same number of features selected by maximal importance criterion. To allow a reliable conversion between the cost and the number of features explored, we assume that all the features have the same cost  $\delta$ . The Interpretable Cascade Classifier with Abstention (ICCA) is expected to have an advantage over the random forests classifier in case where it is reasonable to stop the exploration quite early, or, on the contrary, if an extensive individualized feature exploration is necessary for a particular observation.

In this section, we discuss our results on two standard benchmarks in details. These sets are downloadable from the UCI Machine Learning repository<sup>1</sup> [13]:

- Breast Cancer Wisconsin (Prognostic). We dispose of about 30 parameters describing characteristics of the cell nuclei present in the medical images for 198 patients. All parameters are continuous.
- Mammography. The Mammography set contains 961 observations and 6 variables, and the goal is to predict the outcome of the breast cancer screening, and to avoid unnecessary invasive procedures such as biopsy.

On Figures 2 and 4, we see that both ICCA and the p-ICCA models outperform the random forest classifier in terms of accuracy on the two data sets. The number of features shown on the Figures is a mean over the test sample (5-fold cross-validation) estimated by our model. For a fixed value of  $\delta$  (the unit cost of a feature), the pretrained version of our model (p-ICCA) has a better accuracy than its standard version (ICCA), but the number of features explored is also

<sup>1</sup><http://archive.ics.uci.edu/ml/>

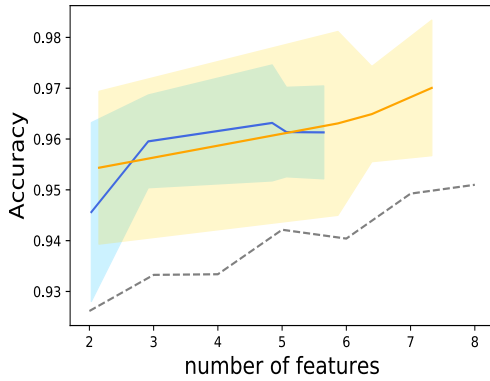


Figure 2: Breast Cancer Wisconsin (Prognostic): Accuracy and standard deviation for ICCA (in blue), p-ICCA (in yellow), and the random forest classifier (dotted line). The features share the same cost:  $\delta \in [10^{-4}, 1/3 \cdot 10^{-2}]$ .

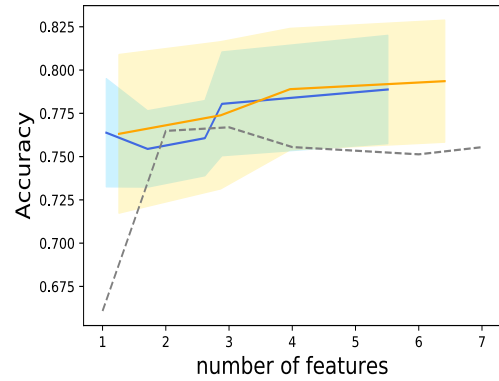


Figure 4: Mammography dataset: Accuracy and standard deviation for ICCA (in blue), p-ICCA (in yellow), and the random forest classifier (dotted line). The features share the same cost:  $\delta \in [3.6 \cdot 10^{-3}, 5.9 \cdot 10^{-2}]$ .

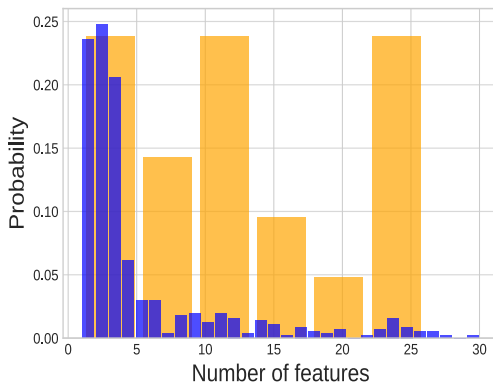


Figure 3: Breast Cancer Wisconsin (Prognostic): histogram of the number of features explored for ICCA. In blue: all observations (size: 569, mean: 4.8); in orange: the misclassified observations (size: 21, mean: 12.3). Overall accuracy: 96,31%.

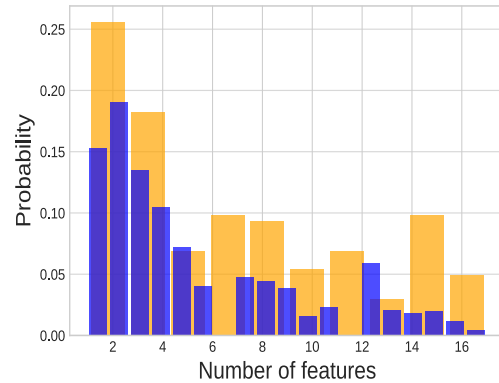


Figure 5: Mammography dataset: Histogram of the number of features explored for ICCA. In blue: all observations (961, mean: 5,5); in orange: the misclassified data (size: 203, mean: 6,7). Overall accuracy: 78,87%.

higher. Thus, for  $\delta$  varying in an interval, the accuracy curves for p-ICCA are quite similar to the ones of the ICCA. The histograms on Figures 3 and 5 represent the distribution of the number of features explored before a classification decision by ICCA. The results have been obtained for a fixed parameter cost  $\delta$  over the test samples produced by a 5-fold cross-validation. In both cases, the distributions are concentrated around their means. For some patients, the method allows to explore a big number of features. The distributions for the misclassified patients show that, in general, the method allows a longer exploration for patients which are difficult to classify. This is especially true for the Breast Cancer Wisconsin data set: for 75% of the misclassified subjects, the number of features explored is

higher than the average on all data (4.8), 40% of the misclassified individuals are associated to a depth in the cascade higher than 15. This is a sensitive result which shows that our model identifies the difficult cases and tries to collect more information by feature exploration.

The results are less clear for the Mammography task. However, the mean of depth in the cascade for the misclassified data is still significantly above the mean of all observations (Figure 5).

Figure 6 shows obtained exploration paths identified by the p-ICCA (unit cost:  $\delta = 1/120$ ) on the Breast Cancer data. The nodes are features ids, and the values on the edges reflect the patients percentage in this or that group. It can be observed that patients are

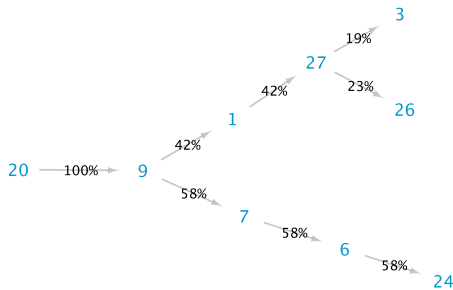


Figure 6: Breast Cancer Wisconsin: The cascade proposed by p-ICCA, exploration path of maximal depth 5 for all patients.

naturally stratified by the algorithm into three clusters (or three paths), and three diagnostic protocols can be developed.

The individualized choice of features reveals some natural questions about personalised diagnostic protocols. The weights of the softmax regression of the classifier  $f_w$  can give some insights in it. If the features 20, 9 and 7 have been observed, the choice of classifying or not can be associated to a strong evidence or a lack of evidence associated to one of the features. This can be explained by studying the difference of weights:  $\beta_{20}^1 - \beta_{20}^0$ ,  $\beta_9^1 - \beta_9^0$  and  $\beta_7^1 - \beta_7^0$ , where 1 and 0 are the classes (see Eq. (13)). However, the choice of a particular path (e.g., feature 7 instead of 1) is less easy to interpret, since it is related to the interaction between the classifier and the selection/rejection model.

We clearly see from the results of the numerical experiments, that the proposed method outperforms the random forest on the Breast Cancer and on the Mammography data sets. On other tested benchmarks of the UCI repository, such as Bankruptcy, Ionosphere, Haberman’s survival, Mushrooms, and the Glaucoma data, our method achieves a very reasonable performance, however, it does not outperform the state-of-the-art. We also concluded that for data sets where there is a couple of dominant features (and they are the same for all observations), our method is not really needed.

**Implementation details:** The neural networks for the rejection/selection model are composed of three fully connected layers. Each layer contains 5 units for the Breast Diagnostic Wisconsin dataset and 4 nodes for the Mammography data set. The activations are the Relu function for the two first layers, and the soft-plus function ( $x \rightarrow \log(1 + e^x)$ ) for the last one. The loss function is the standard MSE.

The neural networks corresponding to the classifier are composed of two fully connected layers, and the last deterministic layer is the softmax regression (14). The loss function is the cross-entropy. For both neural networks, we use the Adam optimizer with the learning rate equal to 0.01.

During the Q-learning procedure, we use an  $\epsilon$ -greedy principle starting with 1, with a decay of 0.995 at each episode. The minibatch sample size for the experience replay memory is 32. At each episode, a new cascade associated to one observation is added to the memory. We run 2000 episodes. We also run 2000 episodes for the pretraining (40 epochs are run for the classifier). All above mentioned hyperparameters of the cascade classifier with abstention have been fixed by cross validation.

Our implementation in Keras will be made publicly available shortly.

## 6 Conclusions

The challenge was to develop a principled approach to learn individual diagnostic protocols under limited budget. The proper theoretical support for the proposed formulation is provided in Section 3. The result of Theorem 1 provides the theoretical explanations for the optimal decision in terms of classification and rejection in a MDP system, and a better understanding of the cases where a prediction has to be postponed, and new data have to be acquired.

An important advantage of the proposed method is that its formalization is simple and implementation is straightforward. Our approach does not compromise the usage of the deep Q-networks. In addition, one could easily incorporate the cost values which can be specific for each separate variable or the cost can be defined for groups of variables.

Currently we are investigating novel solutions based on sparse models to make the deep models interpretable. Another avenue of research are various models for classifiers and reject functions. We are also interested to incorporate Monte Carlo Tree Search into the proposed cascading framework.

## Acknowledgements

The work was supported by the French National Research Agency (ANR JCJC *DiagnoLearn*).

## References

- [1] J. Aron-Wisniewsky, N. Sokolovska, Y. Liu, et al. The advanced-DiaRem score improves prediction



- of diabetes remission 1 year post-Roux-en-Y gastric bypass. *Diabetologia*, 60(10):1892–1902, 2017.
- [2] Peter L Bartlett and Marten H Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(Aug):1823–1840, 2008.
- [3] M. Chen, Z. Xu, K. Weinberger, O. Chapelle, and D. Kadem. Classifier cascade for minimizing feature evaluation cost. In *AISTATS*, 2012.
- [4] C.K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970.
- [5] C. Cortes, G. DeSalvo, and M. Mohri. Learning with abstention. In *ALT*, 2016.
- [6] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *International Conference on Algorithmic Learning Theory*, pages 67–82. Springer, 2016.
- [7] Y. Grandvalet, A. Rakotomamonjy, J. Keshet, and S. Canu. Support vector machines with a reject option. In *NIPS*, 2009.
- [8] B. Hanczar and A. Bar-Hen. Controlling the cost of prediction in using a cascade of reject classifiers for personalized medicine. In *International Joint Conference on Biomedical Engineering Systems and Technologies*, 2016.
- [9] B. Hanczar and M. Sebag. Combination of one-class support vector machines for classification with reject option. In *ECML*, 2014.
- [10] Hideaki Itoh and Kiyohiko Nakamura. Partially observable Markov decision processes with imprecise parameters. *Artificial Intelligence*, 171(8-9):453–490, 2007.
- [11] AG Jones and AT Hattersley. The clinical utility of c-peptide measurement in the care of patients with diabetes. *Diabetic Medicine*, 30(7):803–817, 2013.
- [12] H. Lakkaraju and C. Rudin. Learning cost-effective and interpretable treatment regimes. In *AISTATS*, 2017.
- [13] M. Lichman. UCI machine learning repository, 2013.
- [14] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.
- [15] F. Nan and V. Saligrama. Adaptive classifiers for prediction under a budget. In *NIPS*, 2017.
- [16] F. Nan, J. Wang, and V. Saligrama. Pruning random forests for predicting on a budget. In *NIPS*, 2016.
- [17] S. Nemati, M. M. Ghassemi, and G. D. Clifford. Optimal medication dosing from suboptimal clinical examples: a deep reinforcement learning approach. *Engineering in medicine and biological society*, 2016.
- [18] A. Raghy, M. Komorowski, I. Ahmed, L. Celi, P. Szolovits, and M. Ghassemi. Deep reinforcement learning for sepsis treatment. *arXiv preprint arXiv:1711.09602*, 2017.
- [19] R. Sabbadan, J. Lang, and N. Ravoojanahary. Purely epistemic markov decision processes. In *AAAI*, 2007.
- [20] S.M. Shortreed, E. Laber, D. J. Lizotte, T. Scott Stroup, J. Pineau, and S. A. Murphy. Informing sequential clinical decision-making through reinforcement learning: an empty study. *Machine Learning*, 84:109–136, 2011.
- [21] K. Trapeznikov and V. Saligrama. Supervised sequential classification under budget constraints. In *AISTATS*, 2013.
- [22] P. Viola and M. J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [23] L. Wang, W. Zhang, X. He, and H. Zha. Supervised reinforcement learning with recurrent neural network for dynamic treatment recommendation. In *KDD*, 2018.
- [24] W.-H. Weng, M. Gao, Z. He, S. Yan, and P. Szolovits. Representation and reinforcement learning for personalized glycemic control in spetic patients. *arXiv preprint arXiv:1712.00654*, 2017.
- [25] Y. Zhao, D. Zheng, M. A. Socinski, and M. R. Kosorok. Reinforcement learning strategies for clinical trials in nonsmall cell lung cancer. *Bio-metrics*, pages 1422–1433, 2011.