

Sélection d’espaces de représentation et diagnostic contextuel

André SMOLARZ¹, Pierre BEAUSEROY¹, Yuan DONG¹

¹Institut Charles Delaunay - Equipe Modélisation et Sécurité des Systèmes - UMR CNRS 6279
12 rue Marie Curie, BP 2060, 10010 Troyes Cedex, France
andre.smolarz@utt.fr, pierre.beauseroy@utt.fr, yuan.dong@utt.fr

Résumé – On considère un problème de classification à 2 classes, dans un espace de représentation initial $\mathcal{X}^m \subseteq \mathbb{R}^m$, pour lequel les caractéristiques statistiques de certaines composantes peuvent être altérées selon le contexte. Dans le but de prendre les décisions sur la base de composantes valides nous proposons d’extraire de \mathcal{X}^m un ensemble de sous-espaces de représentation de dimension $d < m$, dans chacun desquels opère un classifieur. La décision finale est prise après fusion des décisions des classifieurs dont le sous-espace associé est non altéré.

Abstract – We consider a classification problem with two-class in an initial representation space $\mathcal{X}^m \subseteq \mathbb{R}^m$, for which the statistical characteristics of some components may be altered due to the context. In order to make decisions on the basis of valid components, we propose to extract a set of representation subspaces of dimension $d < m$ from \mathcal{X}^m , each of which generates a classifier. The final decision is taken by the fusion of these decisions of classifiers which are generated by the unaltered subspace.

1 Contexte de l’approche

Nous considérons le problème du diagnostic de l’état d’un capteur en fonction de l’information délivrée par lui-même et par des « capteurs associés ou voisins ». On admettra dans cette étude qu’un capteur peut présenter uniquement 2 états : bon fonctionnement (classe ω_1) ou panne (classe ω_2). L’état des capteurs « voisins » est susceptible d’évoluer au cours du temps ou selon le contexte (temps, espace, conditions d’exploitation du système...) et par conséquent, la validité ou les performances d’une règle de décision reposant sur les sorties des « voisins » vont également évoluer selon le contexte.

1.1 Formalisme

Considérons un vecteur $\mathbf{X} = [X_1, X_2, \dots, X_m]^t \in \mathcal{X}^m \subseteq \mathbb{R}^m$ où X_i est une variable aléatoire représentant l’attribut délivré par le $i^{\text{ème}}$ capteur appartenant aux $m - 1$ « voisins » du capteur à diagnostiquer (l’attribut du capteur à diagnostiquer fait partie du vecteur). La loi conjointe de \mathbf{X} dépend de l’état des $m - 1$ voisins, que l’on peut résumer selon les 3 situations suivantes :

$$\text{normal} \quad f_{\mathbf{X}}(\mathbf{x} / \omega_1) \quad (1)$$

$$\text{panne} \quad f_{\mathbf{X}}(\mathbf{x} / \omega_2) \quad (2)$$

$$\text{panne de } ncp \text{ capteurs} \quad f_{\mathbf{X}}(\mathbf{x} / \omega_1^{\{m-ncp\}} \cup \omega_2^{\{ncp\}}) \quad (3)$$

Dans le dernier cas (3), le vecteur \mathbf{X} se trouve alors dissocié en deux sous-vecteurs \mathbf{X}_1 et \mathbf{X}_2 de dimensions respectives $m - ncp$ et ncp , dont les modèles conjoints sont

respectivement ceux des classes ω_1 et ω_2 . Le modèle de \mathbf{X} est donc un mélange des modèles décrivant les classes ω_1 et ω_2 et on ne dispose pas de données d’apprentissage pour ce type de situation.

1.2 Règles de décision locales

Afin de disposer d’un système de décision qui reste opérationnel et qui limite la dégradation des performances en cas de panne, nous proposons de décider entre ω_1 et ω_2 en fusionnant les décisions individuelles d’un ensemble de règles dont chacune n’exploite qu’une fraction des mesures provenant du système à surveiller. En effet, si l’on choisit d’utiliser une ou plusieurs règles de décision reposant sur l’espace complet \mathcal{X}^m et apprises sur des données de dimension m , leurs performances seront affectées lorsque l’on se trouvera dans la situation (3) décrite à la section 1.1, puisque cette situation n’a pas fait l’objet d’apprentissage. Il s’agit d’un cas particulier des méthodes d’ensembles [1]. Notre approche consiste donc, dans ce contexte, à écarter les sous-espaces de représentation aux composantes altérées et à ne conserver, dans la mesure du possible, que les règles de décision opérant sur des données conformes à leur apprentissage. Ceci peut s’apparenter à un problème de détection de nouveauté ou d’anomalie [2, 3, 4]. Dans la section 2 nous présentons succinctement le principe de notre méthode, la section 3 présente le contexte et les notations relatifs à l’application sur des images à deux classes de textures ainsi que les résultats obtenus. Pour conclure nous proposerons quelques commentaires et perspectives.

2 Principe

Nous nous sommes inspirés de la méthode *One-class SVM* [5, 6, 7] pour proposer un système décisionnel dans lequel chaque classifieur à une classe possède deux fonctions : valider la règle de décision et décider en faveur (ou pas) de la classe modélisée. Dans notre approche, nous avons considéré deux classes, ω_1 (classe "normale") et ω_2 (classe "panne" ou "anomalie")

2.1 L'approche *One-class SVM*

L'objectif de la méthode (Cf. [5]) est d'estimer une fonction f qui soit positive sur un sous-ensemble S de l'espace de représentation contenant une certaine proportion des données d'une classe et négative sur \bar{S} .

Soit $\mathcal{X}^m \subseteq \mathbb{R}^m$ un espace de représentation de dimension m et un ensemble d'apprentissage $\mathcal{A}_n = \{(\mathbf{x}_i, y_i) \mid i = 1, \dots, n\}$ avec $\mathbf{x}_i \in \mathcal{X}^m$ et y_i label de la classe de \mathbf{x}_i ($y_i = 1$ ou 2). Soit ϕ une transformation non-linéaire dont le produit scalaire de l'image puisse être calculé par un noyau simple $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ (par exemple un noyau gaussien $K_\sigma(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$).

Il s'agit alors de projeter les données dans un espace induit par K puis d'en séparer une proportion $1 - \nu$ de l'origine par l'hyperplan de marge maximale (hyperplan le plus éloigné de l'origine). On peut montrer que cela revient à résoudre le problème suivant :

$$\begin{cases} \min_{\alpha} & \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{sous les contraintes} & 0 \leq \alpha_i \leq \frac{1}{\nu n}, \quad \sum_{i=1}^n \alpha_i = 1 \end{cases} \quad (4)$$

la fonction de décision est alors donnée par :

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^n \alpha_i K(\mathbf{x}_i, \mathbf{x}) - \rho \right) \quad (5)$$

avec $\begin{cases} f(\mathbf{x}) < 0 & \mathbf{x} \in \bar{S} \text{ (rejet)} \\ f(\mathbf{x}) > 0 & \mathbf{x} \in S \text{ (acceptation)} \\ f(\mathbf{x}) = 0 & \mathbf{x} \text{ est un vecteur support} \end{cases}$

2.2 Principe général de notre approche

Soit donc un espace de représentation initial $\mathcal{X}^m \subseteq \mathbb{R}^m$ et une base d'apprentissage \mathcal{A}_n telle que celle décrite à la section 2.1. Afin de prendre les décisions sur la base de composantes non altérées (Cf. section 1), on procède de la manière suivante :

- On extrait à partir de \mathcal{X}^m , L sous-espaces de représentation \mathcal{X}_ℓ^d de dimension $d < m$, par tirage aléatoire de d attributs parmi les m initiaux ($\ell = 1, \dots, L$).
- Pour chaque sous-espace \mathcal{X}_ℓ^d ,
 - on réalise l'apprentissage des fonctions f_ℓ^1 et f_ℓ^2 définies par (5) à l'aide des couples $(\mathbf{x}_i, y_i) \in \mathcal{A}_n$ ($\mathbf{x}_{i,\ell}$ est la projection de $\mathbf{x}_i \in \mathcal{X}^m$ dans \mathcal{X}_ℓ^d).
 - on définit un classifieur h_ℓ sur la base des fonctions de décision f_ℓ^1 et f_ℓ^2

Une première version de classifieur a été proposée dans [8] que nous noterons $h_\ell^{(1)}$. Nous proposons ici une nouvelle définition, notée $h_\ell^{(2)}$, pour laquelle nous avons obtenu de meilleurs résultats en termes de bonne classification. Soit $\mathbf{x}_\ell \in \mathcal{X}_\ell^d$, on définit :

$$h_\ell^{(1)}(\mathbf{x}_\ell) = d_\ell^1(\mathbf{x}_\ell) - d_\ell^2(\mathbf{x}_\ell) \quad (6)$$

$$h_\ell^{(2)}(\mathbf{x}_\ell) = a_\ell f_\ell^1(\mathbf{x}_\ell) + b_\ell f_\ell^2(\mathbf{x}_\ell) + c_\ell \quad (7)$$

$$\text{où } d_\ell^1(\mathbf{x}_\ell) = \mathbf{1}_{f_\ell^1(\mathbf{x}_\ell) > 0} \widehat{P} [f_\ell^1(\mathbf{x}_\ell) > 0 / \omega_1] + \mathbf{1}_{f_\ell^2(\mathbf{x}_\ell) < 0} \widehat{P} [f_\ell^2(\mathbf{x}_\ell) < 0 / \omega_2] \quad (8)$$

$$\text{et } d_\ell^2(\mathbf{x}_\ell) = \mathbf{1}_{f_\ell^2(\mathbf{x}_\ell) > 0} \widehat{P} [f_\ell^2(\mathbf{x}_\ell) > 0 / \omega_2] + \mathbf{1}_{f_\ell^1(\mathbf{x}_\ell) < 0} \widehat{P} [f_\ell^1(\mathbf{x}_\ell) < 0 / \omega_1] \quad (9)$$

a_ℓ, b_ℓ, c_ℓ et les \widehat{P} sont estimés sur un ensemble de données test.

- La décision finale est prise en fusionnant les décisions des L classifieurs selon les formules suivantes :

$$y_{fin}^{(1)} = \text{sgn} \left(\sum_{\ell \in I_n} h_\ell^{(1)}(\mathbf{x}_\ell) + \theta_1 \right) \quad (10)$$

$$y_{fin}^{(2)} = \text{sgn} \left(\sum_{\ell \in I_n} h_\ell^{(2)}(\mathbf{x}_\ell) + \theta_2/n \right) \quad (11)$$

I_n représente l'ensemble des n ($\leq L$) classifieurs tels que $f_\ell^1(\mathbf{x}_\ell) f_\ell^2(\mathbf{x}_\ell) < 0$ (classifieurs qui ne rejettent pas)

θ_1 et θ_2 sont estimés sur un ensemble test

3 Application et résultats

3.1 Texture : notations et modèle

Afin de simuler un ensemble de capteurs dont les sorties sont liées par une loi conjointe, il nous a semblé que la notion de voisinage de pixels utilisée en modélisation des textures pouvait constituer un support applicatif pertinent et adapté à notre approche. Rappelons qu'une texture est une réalisation \mathbf{x} d'un processus stochastique $\mathbf{X} = \{X_s \mid s \in S\}$, X_s étant une variable aléatoire attachée au site s définie généralement sur $\Omega = \{0, 1, \dots, 255\}$. Un pixel sera assimilé à un capteur délivrant un niveau de gris et la démarche présentée à la section 2 va consister à affecter un label à chaque site d'une image constituée de 2 classes de textures. Nous avons choisi comme espace de représentation initial \mathcal{X}^m , celui engendré par les pixels du voisinage carré 5×5 centré sur chaque site ($m = 25$). Les L sous-espaces de représentation \mathcal{X}_ℓ^d sont choisis de dimension $d = 5$ (on tire au hasard 4 pixels parmi les 24 voisins du pixel central qui, lui, contribue à tous les espaces \mathcal{X}_ℓ^d (Cf. figure 1). Les frontières entre les deux classes de textures constituent ici le contexte dans lequel certaines composantes de \mathcal{X}^m ou \mathcal{X}_ℓ^d peuvent être altérées.

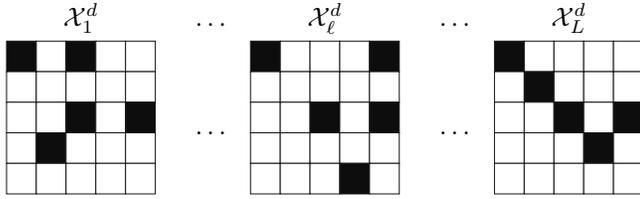


FIG. 1 – Sélection de 4 attributs "voisins" (en noir) parmi 24 (avec les site central, on a donc $m=25$ et $d=5$)

3.2 Exemples traités

Nous avons testé notre approche sur des images constituées de 2 textures. Le premier exemple (figure 4(a)) repose sur les deux textures D21 et D77 tirées de l'album de Brodatz [9]. Les deux exemples suivants (figures 4(b) et 4(c)) reposent sur deux textures markoviennes [10]. Les images des trois exemples sont de tailles 256×256 pixels. Les ensembles d'apprentissage et de test ont, quant à eux, été constitués à partir de vecteurs d'attributs extraits d'images de chacune des 6 textures considérées.

L'image 2 représente la carte des régions (classes ω_1 et ω_2), l'image 3, celle des frontières, constituées d'une bande (en blanc) large de 5 pixels. L'ensemble des pixels "frontières" sert à estimer les probabilités d'erreurs dues aux altérations et son complémentaire permet d'estimer les probabilités d'erreurs de classification dans les "cœurs" des régions (aucun attribut altéré).

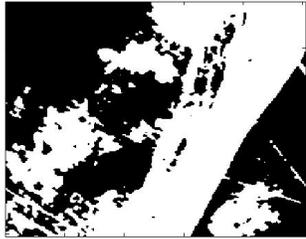


FIG. 2 – Carte des classes



FIG. 3 – Frontières

3.3 Résultats

Les images représentées sur les figures 4(a₁), 4(b₁) et 4(c₁) représentent les classifications obtenues pour chacun des trois exemples, avec le critère de décision $y_{fin}^{(1)}$ défini par la formule (10). Celles représentées sur les figures 4(a₂), 4(b₂) et 4(c₂) représentent les classifications obtenues avec le critère de décision $y_{fin}^{(2)}$ défini par la formule (11). Malgré la faible lisibilité des images en raison du peu de place qui leur est réservé, on peut observer que les résultats de classification sont assez bons, en particulier aux frontières, et ceci pour les deux règles de décision. Afin de juger plus précisément de la qualité des résultats, nous présentons dans le tableau 1, les taux d'erreurs de classification pour chaque exemple. Dans ce tableau, on

distingue les individus étiquetés "frontière" de ceux "étiquetés "cœurs" de régions (Cf. section 3.2). Pour les deux critères de décision reposant sur les One-SVM, nous avons extrait $L = 100$ sous-espaces de dimension 5 selon le principe décrit à la section 2.2. Par ailleurs, afin vérifier notre hypothèse selon laquelle les règles opérant sur l'espace complet \mathcal{X}^m risquent d'être moins performantes en présence d'altérations, nous avons classé nos trois exemples à l'aide d'un SVM à deux classes traitant des données provenant de l'espace complet \mathcal{X}^m ($m = 25$). Les résultats présentés dans le tableau 1, montrent une amélioration significative des performances de notre approche aux frontières entre les régions.

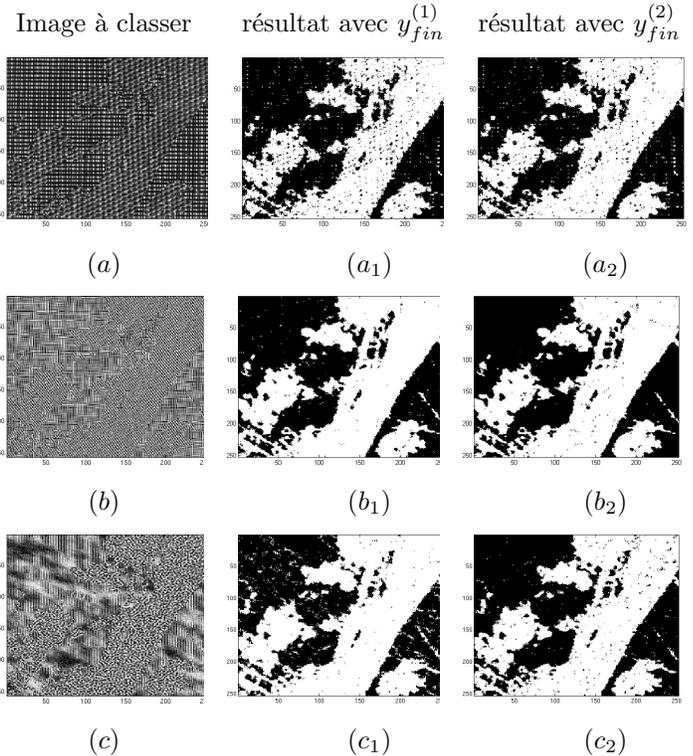


FIG. 4 – Images à classer et résultats de classification

TAB. 1 – Taux d'erreurs (en %) - (ν , σ) donne les valeurs des paramètres des One-SVM

Exemple (ν , σ)	Cœur			Frontières		
	$y_{fin}^{(1)}$	$y_{fin}^{(2)}$	SVM	$y_{fin}^{(1)}$	$y_{fin}^{(2)}$	SVM
(a) (0.15, 40)	3.2	2.2	1.3	17.1	15.8	26.3
(b) (0.1, 75)	0.3	0.3	0.2	10.5	9.1	15.7
(c) (0.1, 100)	2.6	1.1	1.8	21.8	15.5	20.5

4 Discussion

Dans ce papier nous nous intéressons au problème du diagnostic d'un système de capteurs en environnement non stationnaire ou perturbé. Nous avons considéré deux classes, la classe dite "bon fonctionnement" et la classe dite "anomalie" ou "panne".

Les décisions sont prises sur la base de données décrites par un vecteur d'attributs (extraits des capteurs) dans un espace de représentation. L'approche classique consiste, dans ce contexte, à définir et à mettre au point une ou des règles de décision sur la base d'un ensemble d'apprentissage constitué d'exemples étiquetés des deux classes. Toutefois, dans le cas où un certains nombre d'attributs se trouvent altérés, le vecteur descripteur des données représente une situation nouvelle qui n'a pas été apprise.

Pour résoudre ce problème, nous avons donc proposé de prendre les décisions à l'aide d'un grand nombre de sous-espaces tirés de l'espace de représentation initial, espérant ainsi qu'un certain nombre de ces sous-espaces ne contiendront que des attributs altérés ou aucun (les deux situations qui ont fait l'objet de l'apprentissage). La méthode présentée ici repose sur un nouveau critère de décision plus performant que ce que nous avons proposé antérieurement [8]. Par ailleurs nous avons comparé nos résultats avec ceux obtenus par une règle "classique" utilisant un SVM à deux classes opérant sur l'espace de représentation complet de dimension 25. Ces résultats confirment que notre approche s'avère plus performante face aux altérations des attributs.

Dans la suite, nous comptons étudier le lien entre les performances individuelles des classifieurs et la performance obtenue après combinaison, afin d'essayer d'en déduire le meilleur mode de combinaison. Nous envisageons également d'étudier le cas multiclasse.

Remerciements

Ce travail a été réalisé avec le concours du China Scholarship Council (CSC).

Références

- [1] Leo Breiman. Random forests. *Machine Learning*, 45(1) :5–32, 2001.
- [2] Xiaoli Li and Bing Liu. Learning to classify texts using positive and unlabeled data. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 587–594, Acapulco, Mexico, 2003.
- [3] Hwanjo Yu, ChengXiang Zhai, and Jiawei Han. Text classification from positive and unlabeled documents. In *Proceedings of the twelfth International Conference on Information and Knowledge Management*, pages 232–239, New Orleans, LA, USA, 2003.
- [4] Fatih Camci and Ratna Babu Chinnam. General support vector representation machine for one-class classification of non-stationary classes. *Pattern Recognition*, 41(10) :3021–3034, 2008.
- [5] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7) :1443–1471, 2001.
- [6] Paul Hayton, Bernhard Schölkopf, Lionel Tarassenko, and Paul Anuzis. Support vector novelty detection applied to jet engine vibration spectra. In *Advances in Neural Information Processing Systems*, volume 12, pages 582–588, 2000.
- [7] Larry M. Manevitz and Malik Yousef. One-class svms for document classification. *Journal of Machine learning research, Special Issue on Kernel Methods*, 2 :139–154, 2001.
- [8] Pierre Beausery, André Smolarz, and Xiyan He. Décisions en environnement non stationnaire par méthodes d'ensembles via one-class svm - application à la segmentation d'images texturées. In *Proceedings of 42èmes Journées de statistiques*, Marseille, France, 24-28 mai 2010.
- [9] Phil Brodatz. *Textures : A Photographic Album for Artists and Designers*. Dover Publications, New York, 1966.
- [10] André Smolarz. Etude qualitative du modèle auto-binomial appliqué à la synthèse de texture. In *XXIXèmes journées de Statistique*, pages 712–715, Carcassonne, 1997.