Algorithme de calcul des “taux de confiance de référence”

November 14, 2016

Introduction

- Population valide initiale : vecteur ligne de booléen $V \in \{0,1\}^{1\times n}$ qui indique l'état d'un individu au sein de la population : $\begin{cases} 0 & individu\, défaillant\,(invalide) \\ 1 & individu\, sain\,(valide) \end{cases}$,
- Confiance des composants : matrice carrée $M \in \mathbb{R}^{n\times n}$ où le coefficient d'index en ligne i et d'index en colonne j représente le taux de confiance donné par le composant i vis-à-vis du composant j.

1 Principe de fonctionnement

A l'instant initial (décollage d'un vol immédiatement après une maintenance), tout les coefficients du vecteur V sont initialisés à 1 : toute la population d'individus est considérée saine.

L'arbitre cherche à trouver un avis représentatif de la population saine, en “pondérer” l'avis de chaque composant i (vecteur ligne d'index i de la matrice M) par l'avis qu'a toute la population envers ce composant : en effet; si tout les composants font fortement confiance à un composant, il paraît logique que le “poids” de l'avis de composant doit être fortement pris en compte. De même, si les composants font faiblement confiance à un composant, l'avis de ce composant doit faiblement être pris en compte.... Nous obtenons ainsi un avis représentatif de la population, que nous normalisons à 1 en valeur maximale, et que nous appellons “confiances de référence”.

Nous devons empêcher qu'un composant défaillant puisse influencer de façon négative les “confiances de référence” : en effet il se pourrait que plusieurs composants deviennent tour à tour défaillants et, s'ils deviennent majoritaires (et s'accordent entre eux par hasard !), pourraient influencer négativement ces “confiances de référence”, puis finalement invalider les calculateurs valides : nous devons donc éliminer la prise en compte de l'avis d'un composant défaillant dès lors que son comportement anormal est détecté par la population valide.

.

Pour cela, l'arbitre réalise le processus suivant :

- $M \to M_V$: transformation de la matrice M par mise à 0 des lignes d'index i correspondantes à un composant défaillant (i.e. avec i tel que $V(i) = 0$), pour tenir compte uniquement de l'avis des composants sains : $M_V = diag(V) \times M$
- pondération de l'avis de chaque composant j, vecteur ligne d'index j de la matrice M_V, par l'avis qu'ont tout les composants vis-à-vis de ce composant (i.e. la somme des coefficients du vecteur colonne d'index j), puis addition de chacun de ces vecteurs résultant en un vecteur Ω. Si on note $U_{1\times n}$ un vecteur ligne dans $\mathbb{R}^{1\times n}$ et dont tous les coefficients sont égaux à 1, alors on a : $\Omega = U_{1\times n} M_v^2$
- puis normalisation à 1 du coefficient maximal, résultant en un vecteur $\bar{\Omega}$: $\bar{\Omega} = \frac{1}{max(\Omega)}.\Omega$,
- mise à jour du vecteur de population valide : $V \to \bar{\Omega} \geq 0.5$

2 Algorithme

1. **Initialisation** : $V := U_{1\times n}$,
2. **Construction** d'une matrice M de "**taux de confiances**" des **calculateurs**,
3. **Calcul** de la matrice M_V de "**taux de confiances**" des "**calculateurs sains**" : $M_V := diag(V) \times M$
4. **Calcul** du vecteur $\bar{\Omega}$ "**taux de confiance de référence**" : $\bar{\Omega} := \frac{1}{max(U_{1\times n}\times M_v^2)} \times U_{1\times n} \times M_v^2$
5. **Mise à jour** du vecteur V de population "saine" : $V := \bar{\Omega} \geq 0.5$
6. **Retour** à l'étape 2.

3 Code Matlab

Le code Matlab pour implémenter la fonction de décision du superviseur :

```matlab
function [Mref,V] = supervisor_decision_2(M,V)

% Entrée :
%       . M : matrice de "taux de confiances"
%       . V : vecteur de "population saine"
%
% Sortie :
%       . Mref : vecteur de "taux de confiances de référence"
%       . V : nouveau vecteur de "population saine"

n=size(M,1);

% Test de spécification : la matrice M doit être carrée et le vecteur V doit être un vecteur ligne de même taille
assert(size(M,2)==n); assert(size(V,1)==1); assert(size(V,2)==n);

Mv=diag(V)*M;
U=ones(1,n);
K=U*Mv*Mv;

Mref=K/max(K)
V=Mref>0.5;

end
```

4 Exemple concret

Considérons l'exemple suivant :

$$M = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix}, V = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

Ici, le composant 1 accorde un taux de confiance à l'ensemble des composants de $\begin{bmatrix} 1 & 3 & 2 \end{bmatrix}$, le composant 2 un taux de confiance de $\begin{bmatrix} 2 & 1 & 0 \end{bmatrix}$ et enfin le composant 3 un taux de confiance de $\begin{bmatrix} 3 & 3 & 1 \end{bmatrix}$. On peut remarquer que les composants 1 et 3 accordent beaucoup de confiance (valeur de 3) au composant 2. Le composant 2, quant à lui, ne fait pas confiance au composant 3 : on peut donc imaginer assez facilement qu'il ne faudrait pas faire confiance au composant 3. Appliquons l'algorithme et vérifions si notre intuition nous donne raison !

1. **Calcul** de la matrice M_V de "**taux de confiances**" des "**calculateurs sains**" : $M_V \rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix} = M$

2. **Calcul** du vecteur $\bar{\Omega}$ "**taux de confiance de référence**" :

$$U_{1\times n} \times M_v^2 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 13 & 12 & 4 \\ 4 & 7 & 4 \\ 12 & 15 & 7 \end{bmatrix} = \begin{bmatrix} 29 & 34 & 15 \end{bmatrix}$$

$$\bar{\Omega} \rightarrow \frac{1}{max(\begin{bmatrix} 29 & 34 & 15 \end{bmatrix})} \times \begin{bmatrix} 29 & 34 & 15 \end{bmatrix} = \frac{1}{34} \times \begin{bmatrix} 29 & 34 & 15 \end{bmatrix} \approx \begin{bmatrix} 0.85 & 1 & 0.44 \end{bmatrix}$$

3. **Mise à jour** du vecteur V de population "saine" : $V \rightarrow \bar{\Omega} \geq 0.5 = \begin{bmatrix} 0.85 & 1 & 0.44 \end{bmatrix} \geq 0.5 = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

Effectivement, le composant 3 à bien été éliminé par les autres composants : son taux de confiance est inférieur à 0.5. En revanche, le composant 2 lui obtient le meilleur taux de confiance parmi les 3 composants, ce qui correspond bien à ce que nous avions anticipé. Le composant 1 semble lui aussi relativement fiable.

Si l'on reçoit un prochain jeu de données, imaginons pour simplifier la même matrice M, on aura cette fois :

1. **Calcul** de la matrice M_V de "**taux de confiances**" des "**calculateurs sains**" : $M_V \rightarrow diag(V) \times M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 3 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

2. **Calcul** du vecteur $\bar{\Omega}$ "**taux de confiance de référence**" :

$$U_{1\times n} \times M_v^2 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 7 & 6 & 2 \\ 4 & 7 & 4 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 11 & 13 & 6 \end{bmatrix}$$

$$\bar{\Omega} \rightarrow \frac{1}{max(\begin{bmatrix} 11 & 13 & 6 \end{bmatrix})} \times \begin{bmatrix} 11 & 13 & 6 \end{bmatrix} = \frac{1}{13} \times \begin{bmatrix} 11 & 13 & 6 \end{bmatrix} \approx \begin{bmatrix} 0.85 & 1 & 0.46 \end{bmatrix}$$

3. **Mise à jour** du vecteur V de population "saine" : $V \rightarrow \bar{\Omega} \geq 0.5 = \begin{bmatrix} 0.85 & 1 & 0.46 \end{bmatrix} \geq 0.5 = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$

Le composant 3 est à toujours considéré comme défaillant par la nouvelle population "saine" ! L'exemple présenté ici est extrême avec des taux de confiance variés, pas vraiment cohérents et où l'arbitrage est réellement difficile à décider : pourtant le résultat semble cohérent avec notre intuition.