Departement d’informatique

Les Arbres de Decision (Partie 3)

Master 1 MID S2

Enseignant:
Hadjila Fethallah

motivations

- L’objectif principal des algorithmes d’apprentissage est d’améliorer les performances (ex: réduction de l’erreur de généralisation)
- Le travail de [Carvena et al,2006] a comparé 10 classifieurs binaires sur 11 collections de test, en utilisant plusieurs métriques d’évaluation.
- Le classement était:
  - Arbres de décision boostées
  - Random forest
  - Arbre de decision avec bagging (uniquement)
  - SVM
  - Reseau de neurones PMC
  - ….

Methodes d’ensemble

- Principe: entrainer plusieurs classifieurs et utiliser l’union de ces methodes lors de la décision finale (moyenne des resultats individuels ou vote majoritaire)
- Deux classes de methodes ensemblistes
  - Les methode à base de boosting
  - Les methode à base de bagging
- Pour le boosting l’apprentissage se fait seqentiellement, et les classifieurs doivent être faibles « weak learner»(de meme type ou non)
- Pour le bagging, l’apprentissage se fait simultanement, et les classifieurs individuels doivent être de meme type

Random forest

- **Random forest** (foret aleatoire) est un classifieur ensembliste qui utilise plusieurs arbres de decision de type CART.
- Le terme “**random decision forests** ”est inialement proposé Tin Kam Ho of Bell Labs in 1995.
- [Breimans 2001] a proposé la version mature de “**Random forest”** en utilisant deux idées:
  - bagging
  - La selection aleatoire des attributs (random space selection)

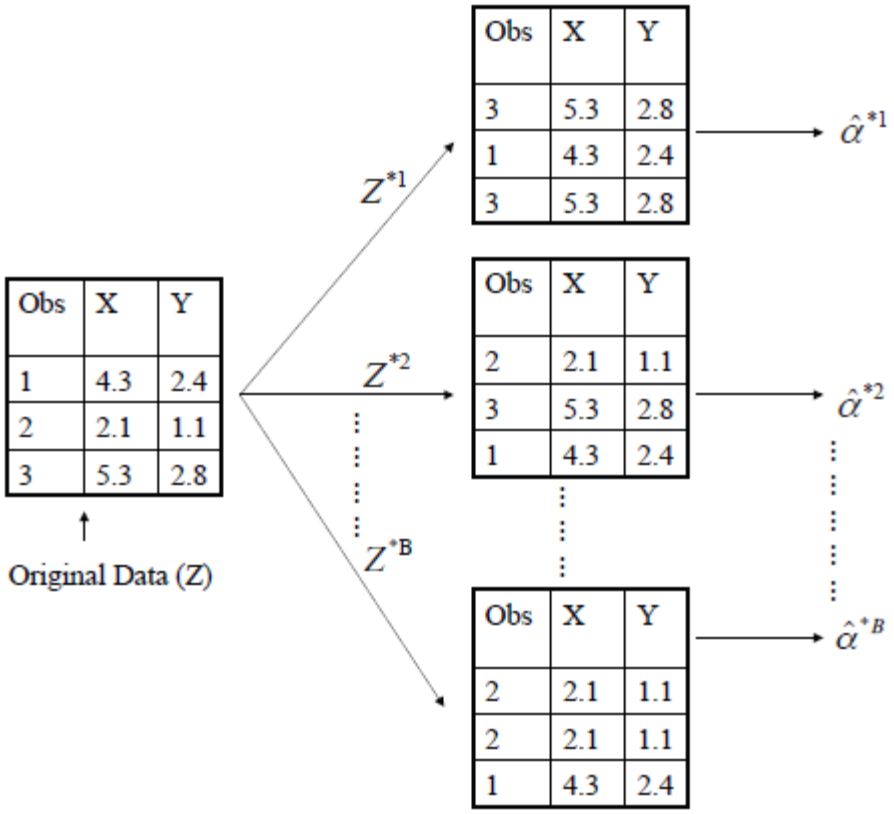
Bagging

- La variance de l'union de B classifieurs (ou modèle de régression) baggés est plus faible que la variance d'un seul classifieur (ou modèle de régression) si on respecte les deux conditions suivantes:
  - Le classifieur (ou modèle de régression) individuel n'est pas biaisé
  - La base d'entrainement Di de chaque classifieur ou (ou modèle de régression) generée à partir de la distribution P du problème de manière indépendante.
- Variance (ClassifieursBaggés)= 1/B variance(classifieur individuel) avec B le nombre de classifieurs
- Intuitivement la variance diminue, si on peut pas mémoriser les exemples, et pour cela nous sélectionnons une partie aléatoire de l'ensemble des lignes et des colonnes
- En plus la moyenne ou le vote majoritaire lisse encore les frontières de décision et de ce fait on diminue la complexité du modele

Bagging

- Bagging = bootstrap aggregation *(Breiman 1996)*
- Echatillon Bootstrap: creer une nouvelle base d'apprentissage en prenant aleatoirement N' exemples de la base originale avec remise (N′≤ le nombre d'exemples de la base originale)
- Bootstrap aggregation:combinaison parallele de B classifieurs entrainés sur des bootstraps differents et de maniere autonome.
- La prediction finale est la moyenne des reponses (regression) ou le vote majoritaire (classification).

exemple


Obs X Y
1 4.3 2.4
2 2.1 1.1
3 5.3 2.8
Original Data (Z)
Z^{*1}
Z^{*2}
Z^{*B}
Obs X Y
3 5.3 2.8
1 4.3 2.4
3 5.3 2.8
Obs X Y
2 2.1 1.1
3 5.3 2.8
1 4.3 2.4
Obs X Y
2 2.1 1.1
2 2.1 1.1
1 4.3 2.4
$\hat{\alpha}^{*1}$
$\hat{\alpha}^{*2}$
$\hat{\alpha}^{*B}$

Random space selection

- L’objectif est de diminuer la corrélation entre les arbres individuels.
- pour cela random forest choisit un sous-ensemble d’attributs de sélection de manière aleatoire pour chaque noeud interne.
- la variance diminue aussi , si on peut pas mémoriser toutes les colonnes de la base
- RS est developpée par Ho 1998, (pour des méthodes homogènes ou hétérogènes, et sans bagging), la valeur par défaut du pourcentage des variables choisies =75%.

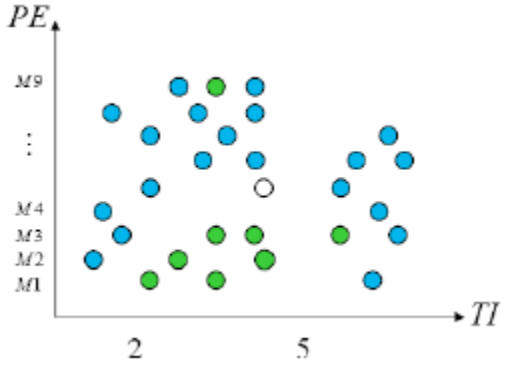
CART

- CART est l’un des arbres les plus utilisés en classification ou regression.
- CART est resumé comme suit
  - Il suit un apprentissage gourmand,
  - Il applique une division binaire et recursive(top-down)
  - Il emploie Gini pour diviser les noeuds (pour la classification)
  - Il emploie L’erreur quadratique pour diviser les noeuds (pour la regression)
  - La decision finale dans une feuille est le vote majoritaire (pour la classification)
  - La decision finale dans une feuille est le la moyenne des sortie des exemples (pour la regression)

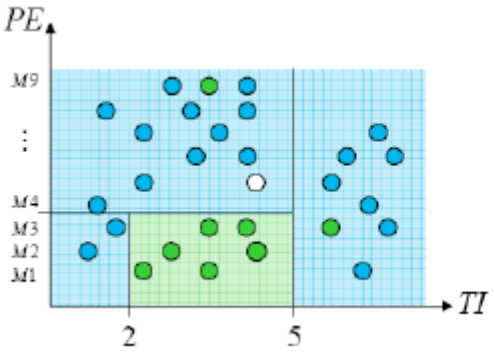
Exemple de deroulement.

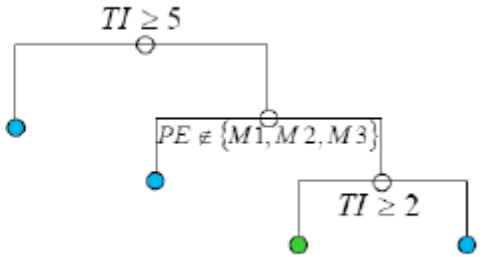
- Une collection de test avec deux attributs

TI	PE	Response
1.0	M2	good
2.0	M1	bad
...	...	...
4.5	M5	?



- Arbre maximal

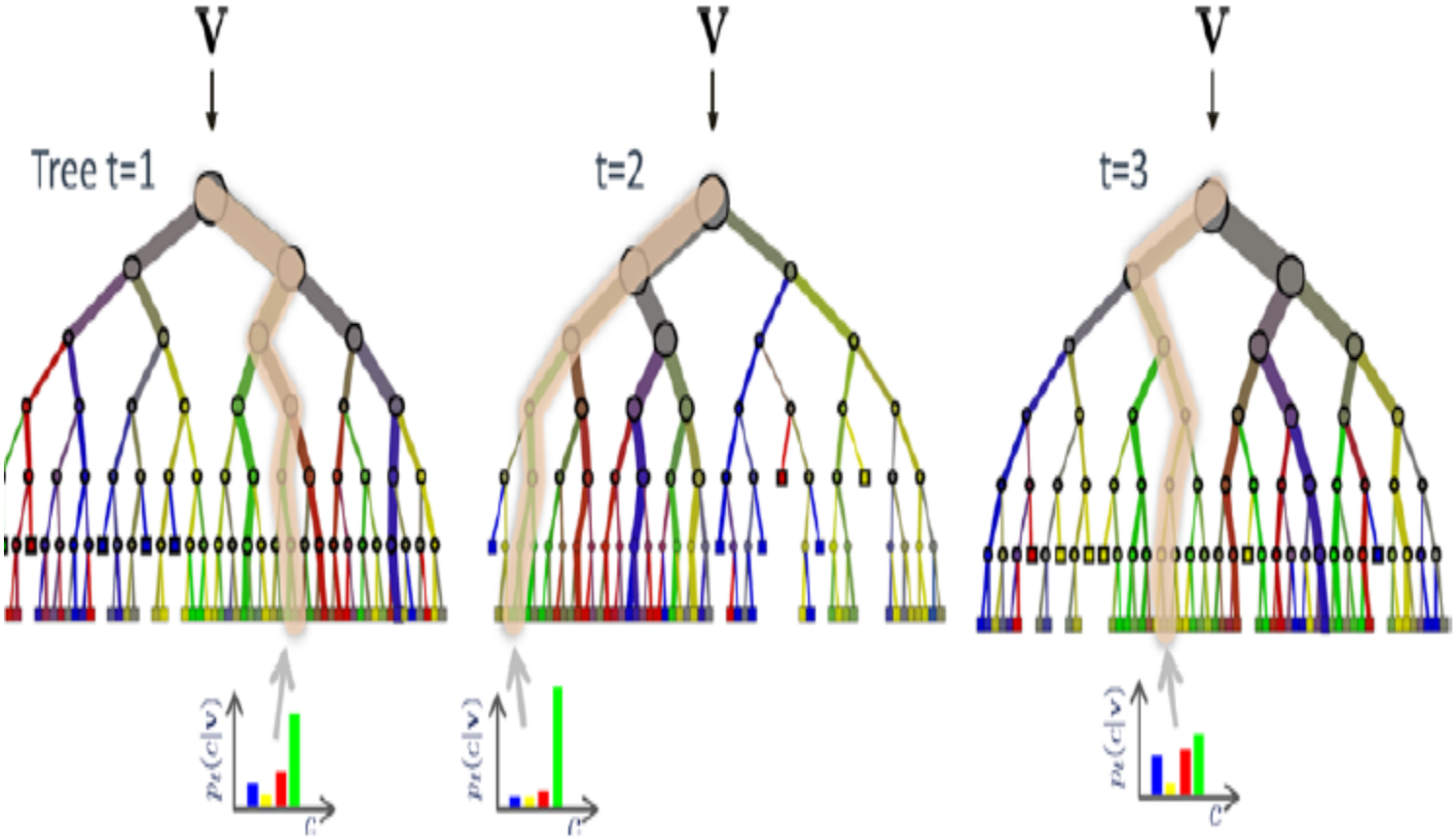




L' algorithme « Random forest »

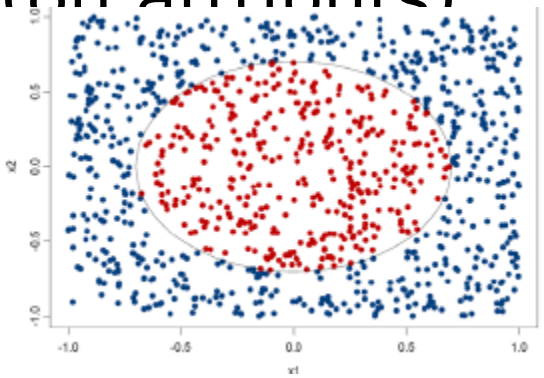
- Soit Ntrees le nombre d'arbres à construire
- Pour ARBRE1 jusqu'à ARBRE*Ntrees*
  - 1. selectionner un sous base "bootstrap" à partir de la base principale (la taille du boostrap=la taille de la base)
  - 2. créer un arbre maximal à partir de ce bootstrap.
  - 3. pour chaque noeud interne, selectionner aleatoirement *m* variables et determiner la meilleure variable de division de ces dernieres.
- La decision finale est
  - la moyenne des reponses des *Ntrees arbres* (pour la regression)
  - le vote majoritaire des *Ntrees arbres* (pour la classification)

L’ algorithme « Random forest »

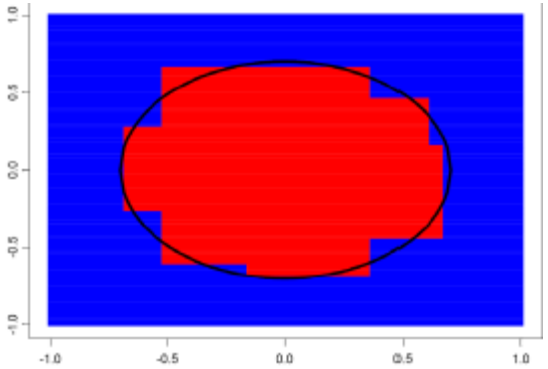


exemple

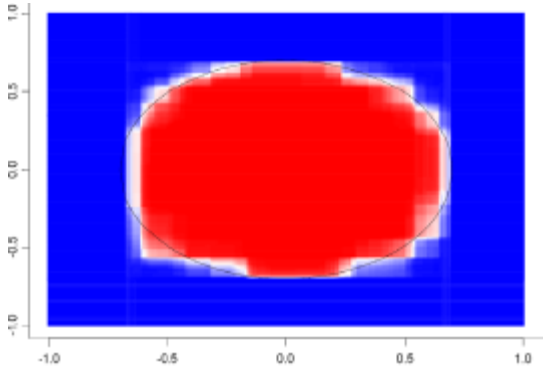
- Base d’exemple originale avec 02 prédicteurs (ou attributs)



Un seul arbre
de decision maximal



100 arbres baggés



ArbreMax vs Random Forest

Test set misclassification error (%)

Data set	Forest	Single tree
Breast cancer	2.9	5.9
Ionosphere	5.5	11.2
Diabetes	24.2	25.3
Glass	22.0	30.4
Soybean	5.7	8.6
Letters	3.4	12.4
Satellite	8.6	14.8
Shuttle $\times 10^3$	7.0	62.0
DNA	3.9	6.2
Digit	6.2	17.1

Collection de test de UCI Repository, table prise de Breiman L, Statistical modeling: the two cultures 2001

Remarques

- Le nombre d'arbre Ntree
  - Ntree est augmenté jusqu'à ce que l'erreur de test stagne(pas besoin de la validation croisée)
- Le nombre d'attributs aléatoires (à chaque nœud):

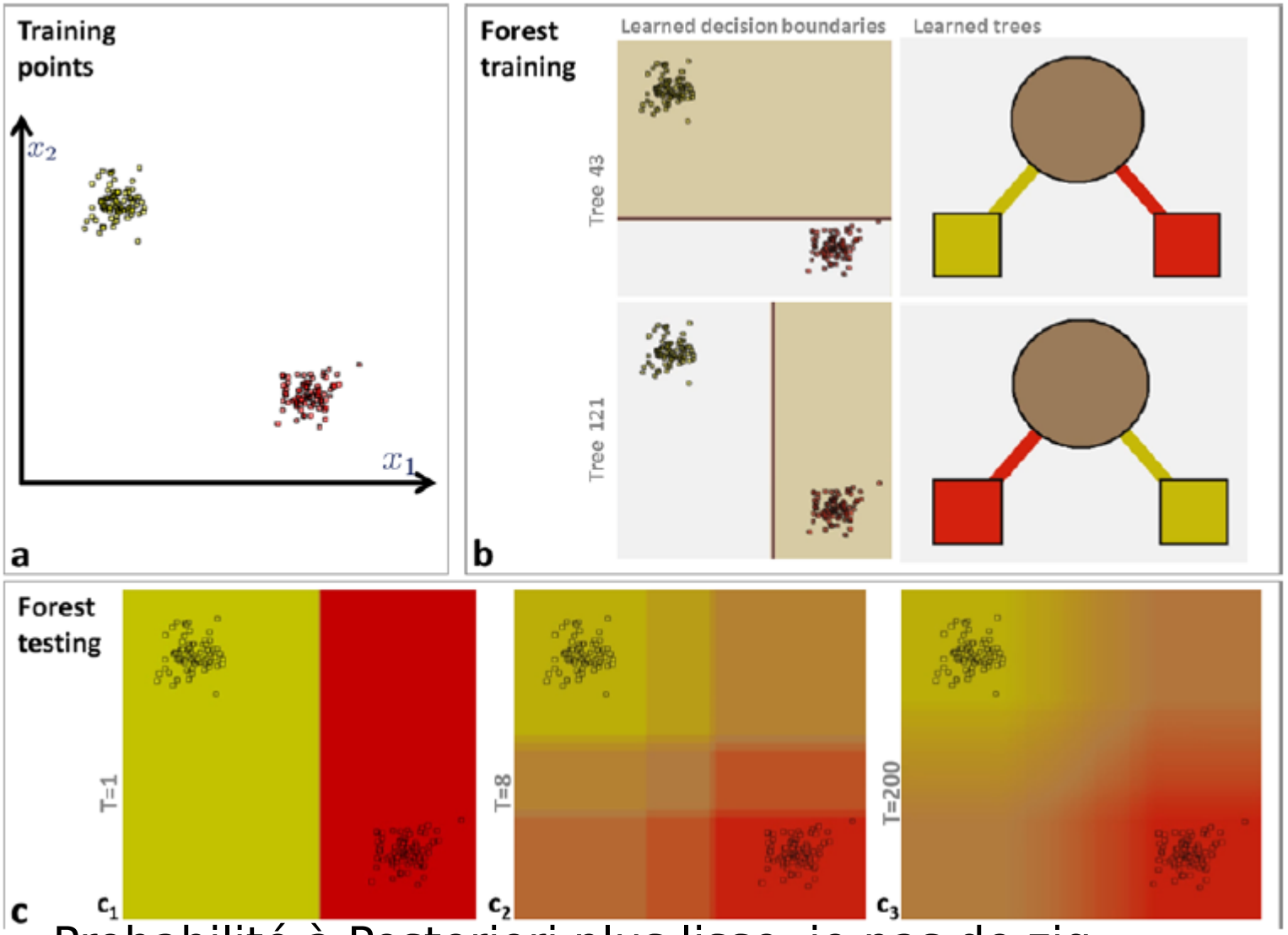
  H1:racine(nbre-variables) pour la calssification

  H2: nbrevariables/3 pour la regression
- Pour chaque arbre, 63,2% des exemples du boostrap sont uniques et le reste c'est des doublons → 1/3 de la base originale n'est pas selectionné dans le bootstrap, il sont nommés out of bootstrap ou out of bag (OOB)
- On utilise les OOB pour estimer l'erreur de test de chaque classifieur. Et eventuellement pour faire un vote ponderé, et meme pour estimer l'importance d'une variable de division.
- Taille du noeud minimale=1 (classification) et 5 pour la regression(Breiman 2001)

L’erreur OOB pour une forêt

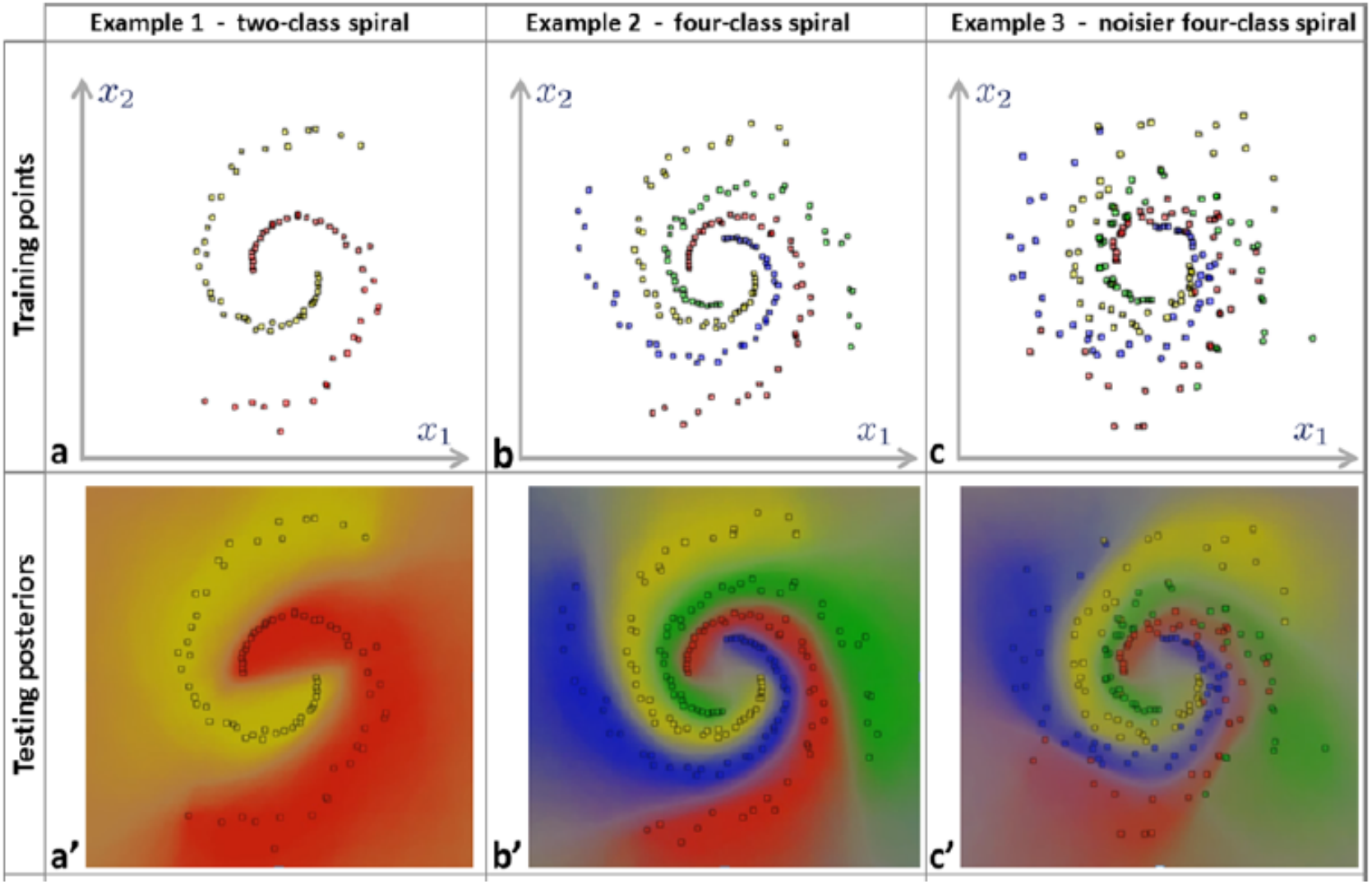
- OOB est un element non selectionné dans le bootstrap
- La prediction faite sur un OOB noté xi est :
- Prediction_OOB(Xi)= 1/|Si| $\sum_{T \in Si}$ T(Xi)
- Si = {Ti ∈ forêt-aleatoire / boostrap(Ti) ne contient pas Xi}
- l’erreur OOB= 1/|base| . Nombre (Xi) ayant Prediction_OOB(Xi)≠ etiquete(Xi))
- l’erreur OOB est une bonne estimation de l’erreur de test.
- Si B est assez grand, l’erreur OOB est similaire à la validation croisée

Effet du nombre d'arbres



Probabilité à Posteriori plus lisse, ie pas de zig zag

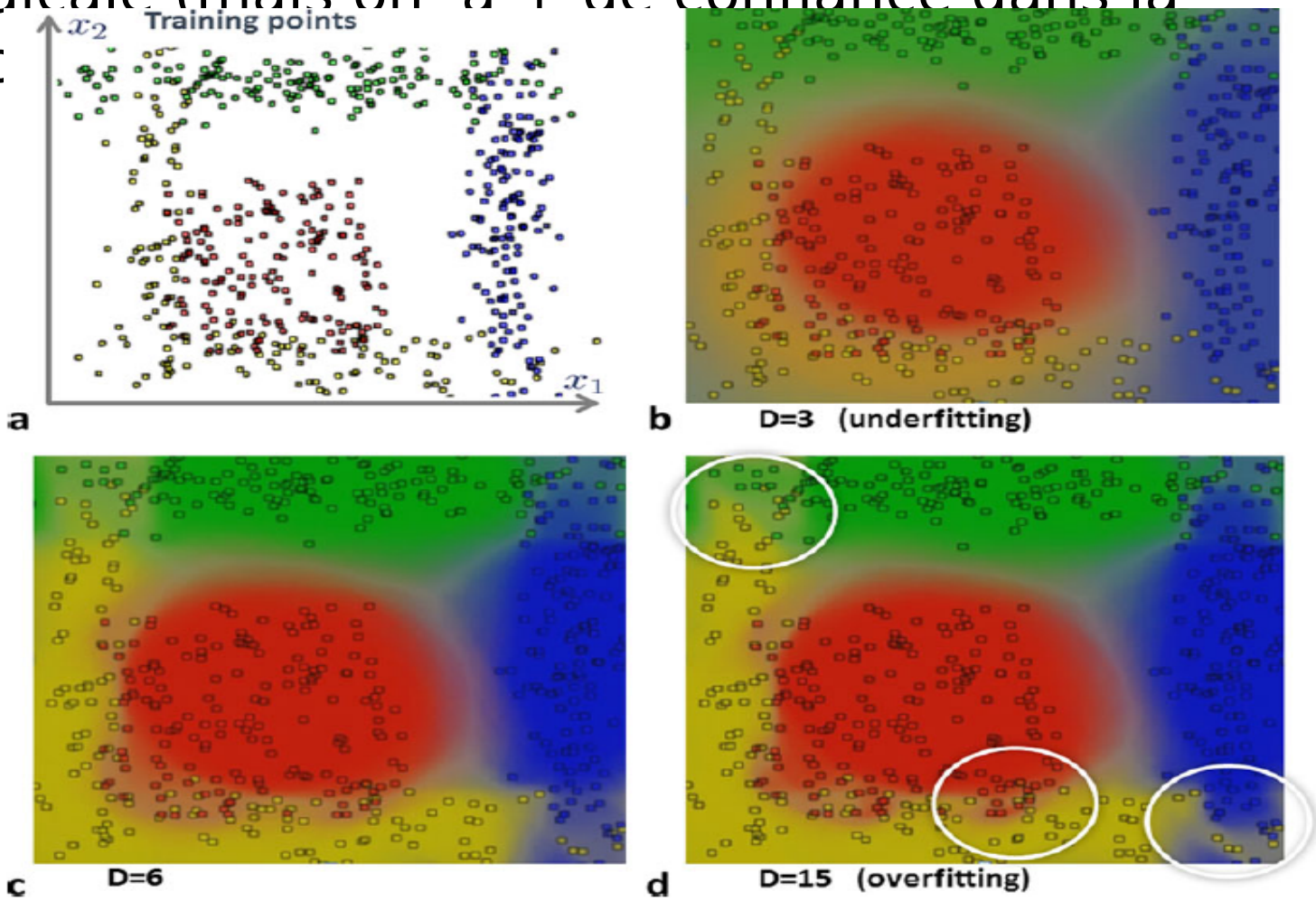
Effet du bruit (données mal-étiquetées)



Incertitude plus grande (faible confiance) car les feuilles sont melangées

Profondeur de l'arbre D

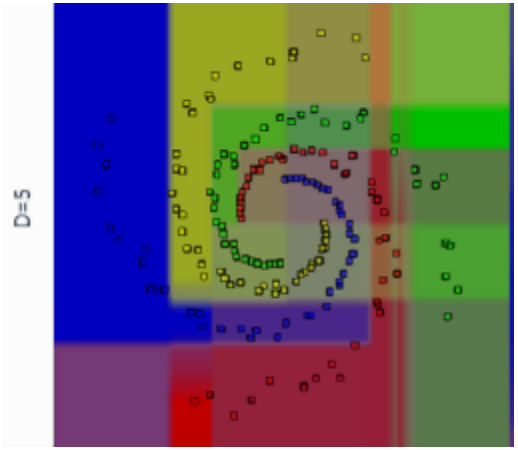
- La profondeur de l' arbre D, plus elle est grande plus le risque du sur apprentissage est fort, et par conséquent la moyenne n'est pas une solution radicale (mais on a + de confiance dans la prc

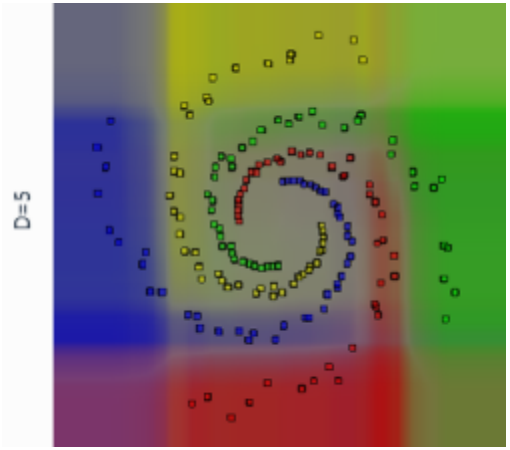


effet de l'aléatoire sur les attributs (randomness)

- Lorsqu'on diminue le nombre d'attributs (et seuils) qui participent dans la competition:
- On diminue la corrélation des arbres
- On Reduit le phénomène de blocs (réduit l'inconsistance ou le Zig Zag)→ Probabilité à posteriori assez lisse
- On aura une confiance affaiblie (plus d'incertitude) car il se peut que les attributs choisis ne sont plus discriminants.
- L'aléatoire fort est très souhaité dans les bases bruitées, car l'influence d'un seul point sur la position de la coupe est trop faible

effet de l‘aléatoire sur les attributs (randomness)

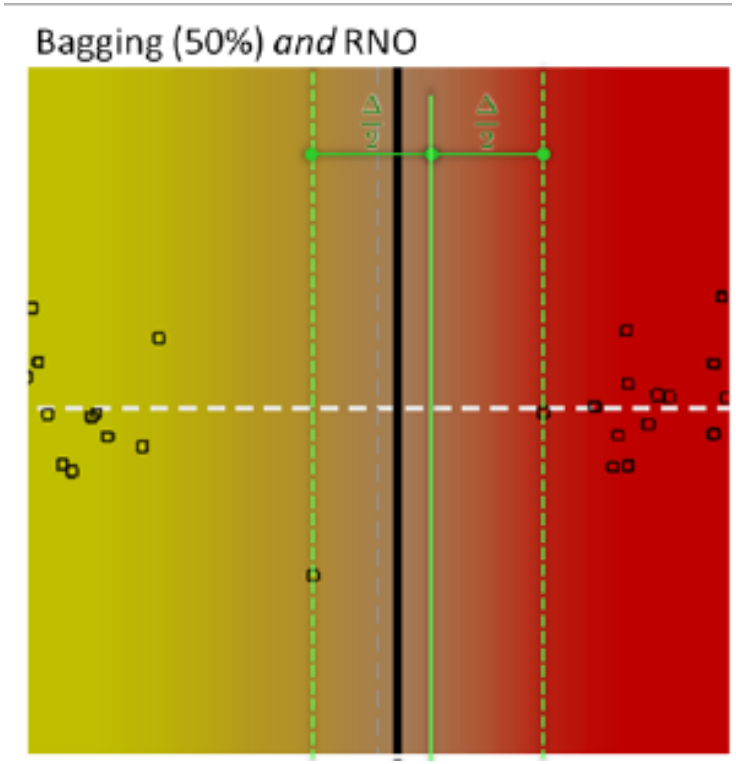




- 500 possibilités de coupes 05 possibilités de coupes

Effet du bagging

- Evitement des outliers lors des coupes

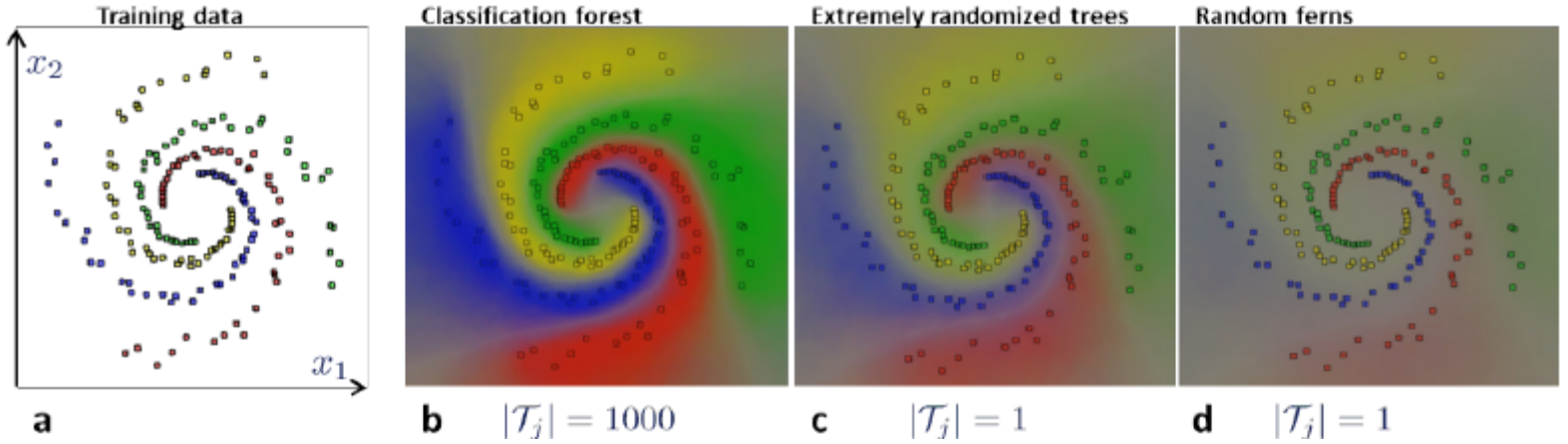


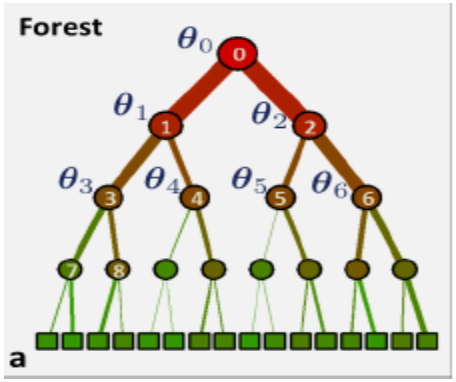
ERT ,PERT, Random Ferns

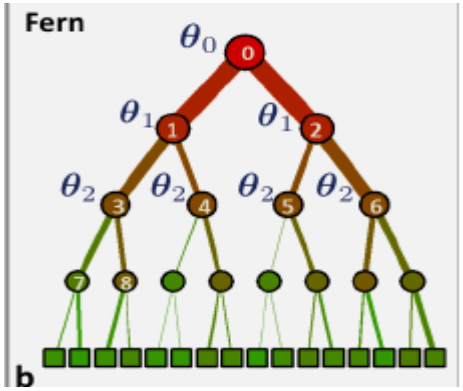
- Pert est ensemble d'arbres de décision (baggés ou non), chacun d'eux est créé en choisissant des variables et des seuils aléatoires (le seuil appartient à un intervalle dont les bornes sont des caractéristiques des points du nœud courant)
- Randomized C4.5: à partir des 20 premières variables de divison on sélection une de manière aléatoire.
- ERT:
- pas de bagging
- Dans chaque nœud, on sélectionne K variables de manière aléatoire, pour chacune d'elles on choisit un seuil aléatoire. pour ces K couples on sélectionne la meilleure paire(variable, seuil) selon la mesure du gain d'information (dans le cas extreme k=1 ie totally R T)
- Random ferns : cas particuliers de RF, ou le même couple (variable seuil) doit être utilisé dans tous les nœuds du même niveau

ERT ,PERT, Random Ferns

- Ces variations sont très rapides en termes d'apprentissage
- Utiles lorsque les exemples sont peu







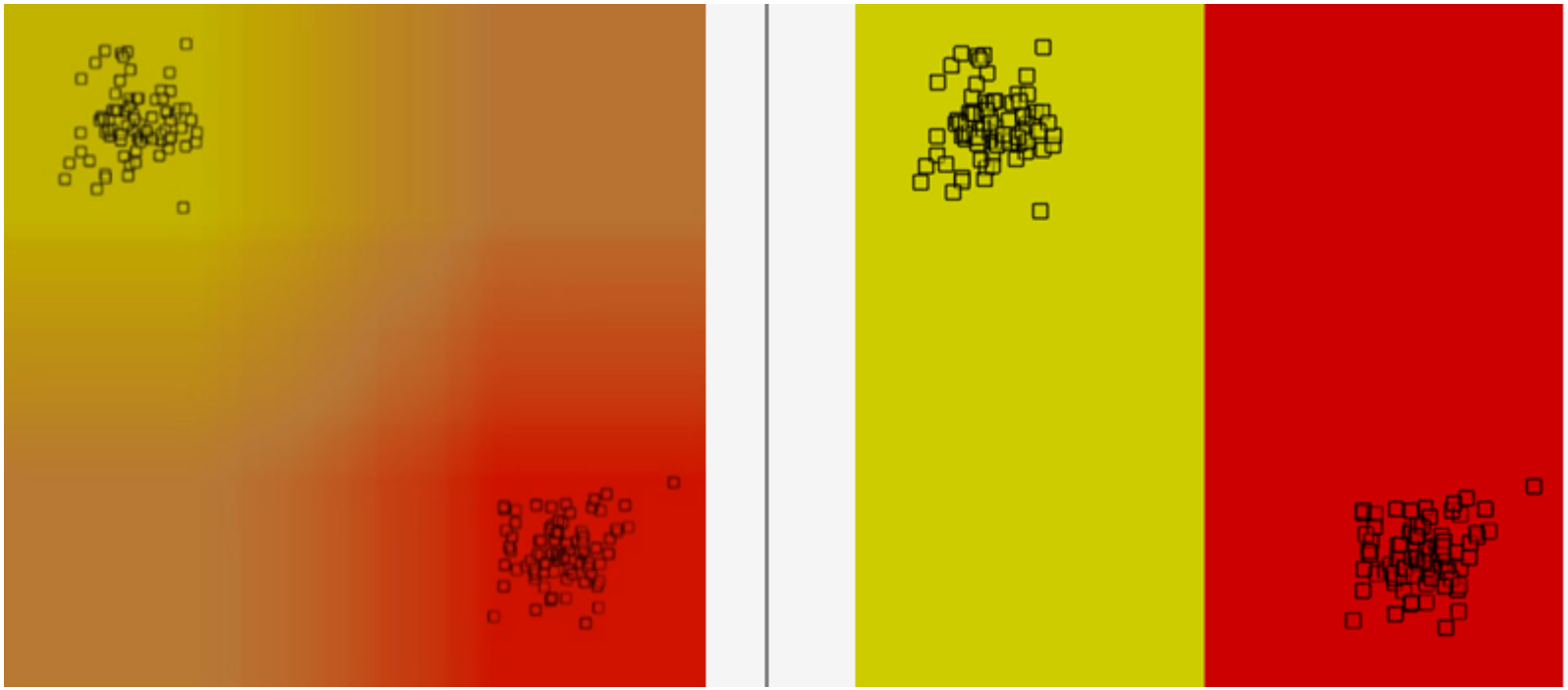
clustering

- Approche 1:
- Importance de la variable V, est la somme des améliorations du Gini associé à V sur tous les éléments de la foret.
- approche 2:
- Pour évaluer l'importance de V, on mesure l' erreur OOB pour chaque arbre T de la foret (ce score est noté ER1)
- On permute les valeurs de V qui concernent les éléments OOB, et on mesure pour la 2eme fois l'erreur OOB de T. (ce score est noté ER2)
- Si V est importante alors 1/|foret| $\sum_{T \in foret}$ (ER2-ER1) est grande et vice versa

avantages

- Il minimise la variance tout en gardant le biais inchangé→ amélioration de l'erreur de prédiction
- Très efficace sur les grande collection surtout lorsque le nombre de variable est grand (ex: identification des relations entre les maladies et les gènes)
- Possibilité du parallélisme
- Il permet de faire aussi la sélection de variables d'entrées
- Il peut traiter les données manquantes
- Il peut traiter efficacement les collection de test non balancées
- Le out of bag sert à estimer l'erreur de généralisation
- Pas besoin de prétraiter les entrées (ex normalisation…)
- Il peut etre utilisé dans le clustering, la détection des outliers

- Des expérimentations comparant le boosting et forets aléatoires en utilisant les decision stumps montre que:
- La frontiere (separant les classes)en boosting est regide alors qu'elle se degrade de manière lisse dans les F.A



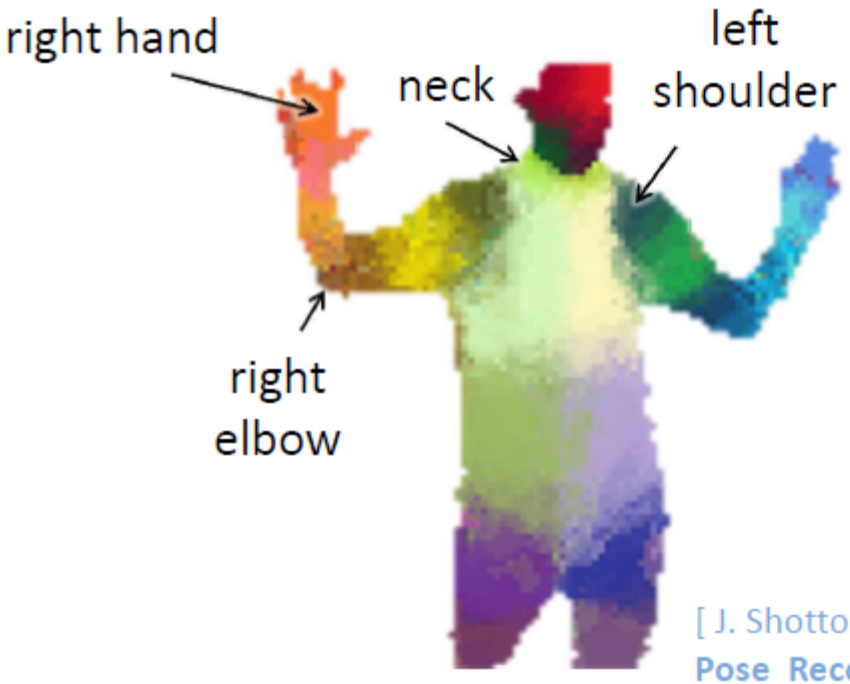
inconvenients

- Si la base est trop bruitée il peut faire du sur apprentissage
- Pour les variables nominales :Il peut favoriser celles qui ont un grand nombre de valeurs .
- Comme cart, random forest ne peut pas donner une prédiction au delà des valeurs appartenant à la base d’apprentissage

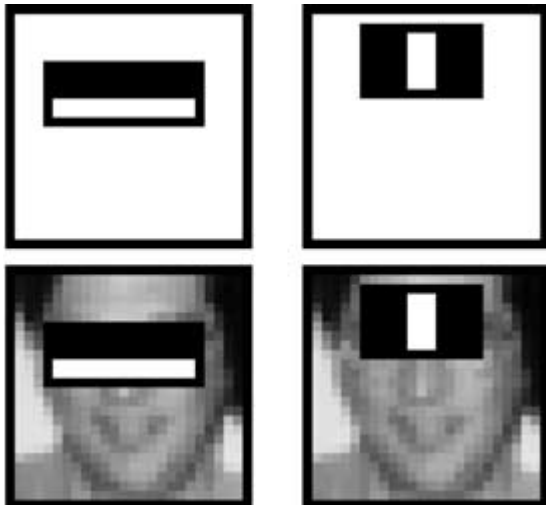
applications

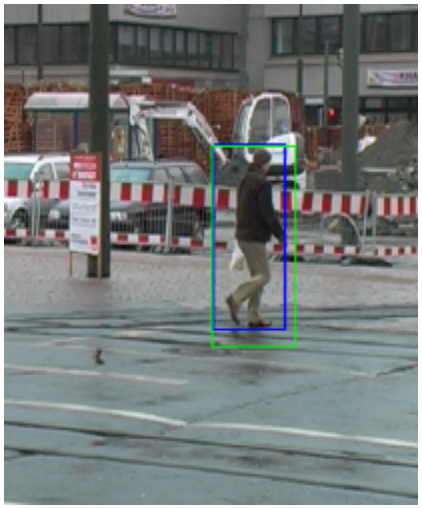
- Localisation des points d’interets

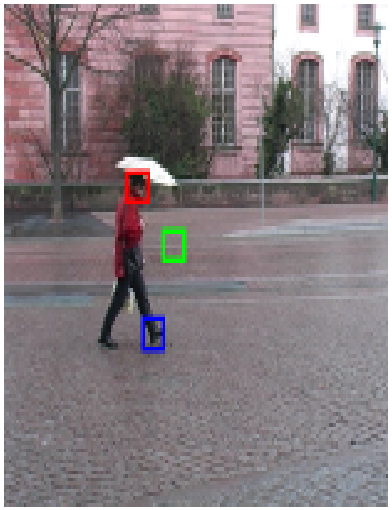
Segmentation des parties du corps

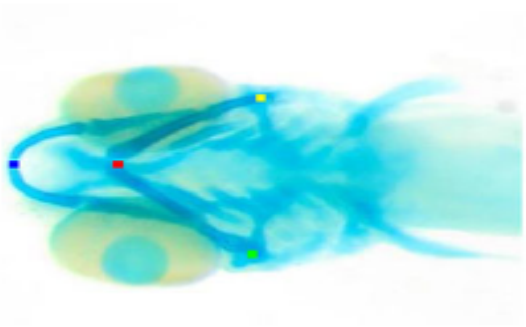


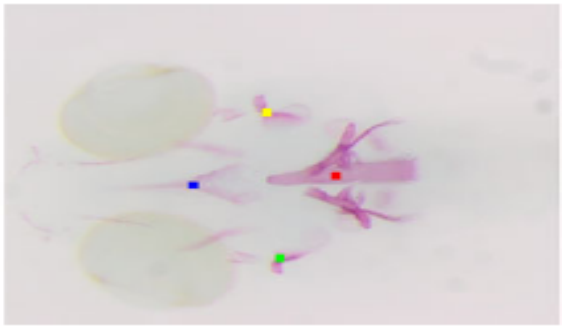
[J. Shotton et al. **Real-Time Human Pose Recognition in Parts from a Single Depth Image**. CVPR 2011]











conclusion

- RF est rapide en termes de construction et de prediction
  (l'echantillonnage des exemples et des variable, parallelisme)
- Resistance au sur apprentissage
- Utilisable pour la sélection de variable, le clustering
- Interpretabilté du modele
- Traitement de données manquantes
- Pas de pre-processing de données ou de validation croisée

References

- 1984, Breiman L, Friedman J, Olshen R, Stone C, Classification and Regression Trees; Chapman & Hall; New York
- 1996, Breiman L, Bagging Predictors. Machine Learning 26, pp 123-140.
- 2001, Breiman L, Random Forests. Machine Learning, 45 (1), pp 5-32.
- 2001, Breiman L, Statistical modeling: the two cultures, Statistical Science 2001, Vol. 16, No.3, 199-231