# Un algorithme d'optimisation à haute dimension pour la fouille de données

## Méthode et application en onco-pharmacogénomique

Vincent Gardeux, René Natowicz, Rachid Chelouah, Roman Rouzier, Antônio Braga Padua, Patrick Siarry

L@ris, EISTI
LiSSi, Université Paris-Est
Université Paris-Est, ESIEE-Paris
Hôpital Tenon

ROADEF 2011 - 2,3 et 4 Mars 2011

# Sommaire

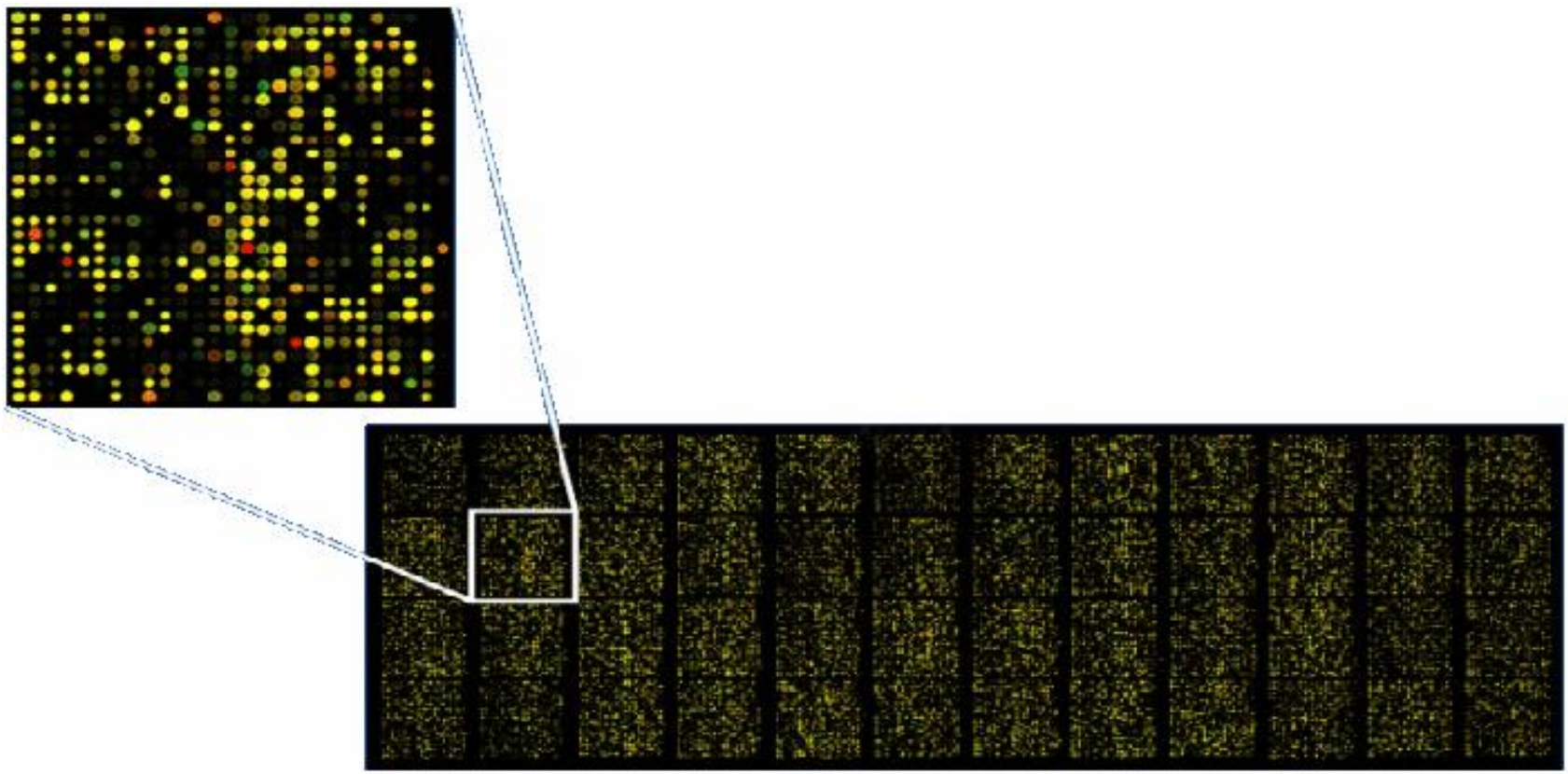
# Fouille de données

La fouille de données ou *Data Mining* permet d'extraire la connaissance à partir d'un ensemble de données.
On peut la décomposer en plusieurs phases :

- *Feature Selection* : Phase de sélection des variables les plus intéressantes (les variables apportant le plus d'information pour la future classification).
- Classification :
  - Apprentissage : On calcule à partir des variables sélectionnées, les règles qui nous permettront de dire à quelle classe appartient un nouvel individu
  - Validation : On estime les performances de notre classifieur

# Puce à ADN

- Données fournies par des puces à ADN
- Fournissent le niveau d'expression d'un ensemble de gènes pour un patient donné



# Données

- Patientes atteintes du cancer du sein
- 2 sources de données utilisant la même puce à ADN (Affymetrix U133A) de 22283 gènes
  - Houston : 82 Patients
  - Villejuif : 51 Patients
- 2 classes de patientes : pour chaque patiente, on sait si elle a été répondeuse (PCR : pathologic complete response) ou non (No-PCR) au traitement chimiothérapique
- Données importantes : $\simeq$ 20000 gènes pour très peu de cas : $\simeq$ 100 patients
- But : Prédire la classe d'un nouveau patient en fonction des niveaux d'expression de ses gènes.

# Méthodes

- On utilisera des procédés pour augmenter virtuellement le nombre de cas d'études (cross-validation, bootstrapping, ...)
- On prend le parti de se focaliser sur l'étape de *Feature Selection*
- Après sélection d'un sous-ensemble de gènes par notre méthode, on utilisera des classifieurs existants pour la création du modèle de prédiction : LDA, DLDA, SVM, ... (disponibles dans Matlab, R, ...)

# Normalisation des données

## Normalisation Logarithmique

- Normalisation nécessaire quand les variables peuvent prendre des valeurs sur plusieurs décades (typiquement $> 5$ décades)
- Utile pour ramener les données sur une échelle commune

## Méthode

But : Normaliser la variation des probes pour toutes les patientes. Soit $x = \{x_1, ..., x_i, ..., x_n\}$ le vecteur à normaliser, et $x_{max}$ et $x_{min}$ ses extrema. Chaque composant $\dot{x}_i$ du vecteur normalisé $\dot{x}$ est calculé ainsi :

$$\dot{x}_i = \frac{log(x_i) - log(x_{min})}{log(x_{max}) - log(x_{min})} \quad (1)$$

avec *log* l'opérateur logarithmique décimal

# Algorithme de *Feature Selection* - Etat de l'art

- La plupart des méthodes de *Feature Selection* actuelles se basent sur des méthodes statistiques pour sélectionner les variables les moins redondantes et apportant le maximum d'information (Student t-test, R-squared, ...)
- Il existe également des techniques d'optimisation (LASSO, algorithmes génétiques, ...), mais la plupart utilisent tout d'abord un test statistique afin de limiter la dimension de l'espace de recherche.

# Algorithme de *Feature Selection* - But

- Sélectionner le sous-ensemble de gènes le plus représentatif :
  - Maximiser la distance entre les 2 classes de patientes (distance interclasse)
  - Minimiser la taille du sous-ensemble de gènes sélectionnés, afin d'éviter le sur-apprentissage des données (*data overfitting*)

  $\Rightarrow$ Optimisation bi-objectif

  $\Rightarrow$ Les 2 objectifs sont contradictoires
- Fonction objectif comme combinaison linéaire convexe de ces 2 objectifs, pondérés par un paramètre $w$ ($w \in [0, 1]$)

$$F_w(s) = w * d(G(s), G'(s)) + (1 - w) * (1 - |s|) \quad (2)$$

# Algorithme de *Feature Selection* - Définitions

**Distance interclasse :**

Soient G(s) et G'(s) les deux vecteurs centres de gravité des deux classes répondeuses et non répondeuses pour le sous-ensemble de sondes s. La distance interclasse est la distance euclidienne entre les deux centres de gravité G(s) et G'(s)

- La fonction bi-objectif est à optimiser dans l'ensemble des sous-ensembles de sondes (ensemble de taille $2^{22283}$)
- Une solution est un vecteur binaire de dimension 22283

*Ex :*

$x = \{0, 1, 0, 0, 0, 1, 0, ..., 1\}$

$\Rightarrow$ Un 1 en position $i$, signifie que le $i^{eme}$ gène est sélectionné

# Algorithme de *Feature Selection* - Méthode

- Problème d'optimisation haute dimension : nous nous sommes basés sur l'algorithme EUS développé pour les problèmes d'optimisation à grande dimension

## Principe de EUS

- Méthode de relaxation pour décomposer la fonction objectif dimension par dimension
- Line Search pour optimiser la fonction objectif sur chaque dimension
- Procédure de redémarrage pour explorer l'espace de recherche

- Discrétisation de l'algorithme EUS (DEUS) $\Rightarrow$ Application sur un vecteur binaire
- Algorithme résultant : suite de minimisations locales
  - Gène retiré ou ajouté au sous-ensemble si cela augmente la valeur de la fonction objectif
  - On recommence jusqu'à stabilisation

# Algorithme de *Feature Selection* - DEUS pseudo code

**Procedure DEUS**
**begin**
Initialiser le vecteur binaire $S$ aléatoirement
**do**
$v = f(S)$ (avec $f$ la fonction objectif)
**for** $i = 1$ **to** $n$
permuter la $i^{eme}$ valeur binaire de $S$
$v' = f(S)$
**if**(v' $<$ v) **then** permuter à nouveau la $i^{eme}$ valeur de $S$
**end for**
**while** aucune meilleure solution n'est trouvée durant une itération complète
**end**

# Protocole Expérimental

- Comparaison avec 2 méthodes récentes :
  - Hess 2006 (t-test + DLDA)
  - Natowicz 2008 (BI Majorité 30 + DLDA)
- Nous avons suivi le protocole utilisé par Hess et Natowicz :
  - Données Houston : Population d'apprentissage (pour la sélection des gènes et pour la méthode de classification)
  - Données Villejuif : Population de validation (pour la méthode de classification)
- Chaque population contient 1/3 de patient répondant au traitement et 2/3 non répondant
- Méthode de classification utilisée : DLDA (*Diagonal Linear Discriminant Analysis*)

# Résultats comparatifs

| | DEUS DLDA 31 | DEUS DLDA 11 | t-test DLDA 31 | BI Majorité 30 |
|---|---|---|---|---|
| Distance interclasse | 5175,45 | **3670,45** | 1383,30 | 3210,28 |
| Précision | 0,863 | **0,882** | 0,765 | 0,863 |
| Sensibilité | 0,846 | **0,923** | 0,923 | 0,923 |
| Spécificité | 0,868 | **0,868** | 0,711 | 0,842 |
| VPP | 0,688 | **0,706** | 0,522 | 0,667 |
| VPN | 0,943 | **0,971** | 0,964 | 0,970 |

- Sensibilité : Probabilité d'être classifiée répondeuse si elle l'est
- Spécificité : Probabilité d'être classifiée non répondeuse si elle l'est
- VPP : Valeur Prédictive Positive = probabilité d'être répondeuse si classifiée ainsi
- VPN : Valeur Prédictive Négative = probabilité d'être non répondeuse si classifiée ainsi

# Gènes sélectionnés

Les 11 gènes sélectionnés par DEUS sont les suivants :

| Sonde Affymetrix | Nom du Gène | Hess | Natowicz |
|---|---|---|---|
| 203929_s_at | **MAPT** | X | |
| 204825_at | MELK | X | X |
| 204913_s_at | SOX11 | | |
| 205225_at | **ESR1** | | |
| 205354_at | GAMT | X | |
| 205548_s_at | **BTG3** | X | X |
| 209173_at | AGR2 | | |
| 212956_at | AI348094 | | |
| 213134_x_at | **BTG3** | X | X |
| 218211_s_at | MLPH | | |
| 219051_x_at | METRN | X | X |

# Conclusion

- Nos résultats surpassent les autres, mais le résultat le plus intéressant est qu'ils sont obtenus avec 3 fois moins de probes (11 au lieu de 30)
- Rapidité de convergence : quelques itérations suffisent
- Solution stable :
  - Ne dépend pas de la solution initiale
  - Ne dépend pas de l'ordre dans lequel les gènes sont examinés
  - A chaque redémarrage, converge vers la même solution pour un $w$ donné
- La Sensibilité est élevée $=>$ important pour les biologistes et éventuellement pour la mise en place d'une routine clinique
- Une Cross-validation 3-fold sur l'ensemble des 133 patientes obtient d'excellents résultats $=>$ Accentue la robustesse de la méthode

# Travaux en cours

- La Méthode Lasso "regularisation path via coordinate descent" (Trevor Hastie & al.) retourne des prédicteurs ayant plus de sondes et dont les performances sont inférieures.
- Fonction objectif intégrant inertie interclasse et inertie intraclasse
- Fonction objectif intégrant la corrélation entre les gènes
- Application à d'autres domaines d'application

Merci pour votre attention

Des questions ?

Vincent Gardeux
Enseignant-chercheur EISTI
Doctorant de l'Université Paris-Est Créteil

Davantage d'informations à propos des recherches en cours :
http ://gardeux-vincent.eu