



## XGBoost, origines et applications

Damien DOUTEAUX

## Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

Sommaire

Aspects théoriques

Mise en œuvre

Applications

Conclusion

## Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⦿ Moteurs de recherche



- ⦿ Alertes mails



- ⦿ Réseaux sociaux



Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

XGBoost : *EX*treme *G*radient *Boo*sting

- ⊙ **Flexibilité** Régression, classification,...
- ⊙ **Portabilité** Windows, Linux, OS X
- ⊙ **Multi-langages** Python, R, JAVA, C++, Scala,...
- ⊙ **Distribué** Yarn, Spark, Flink, AWS, Azure,...
- ⊙ **Performance** Optimisé et expensif

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

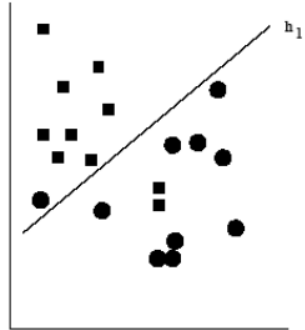
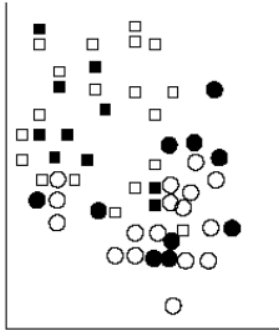
Conclusion

- ⊙ Une stratégie adaptative.
- ⊙ Convertir des règles peu performantes en (très) bonne prédiction.
- ⊙ Réduction variance et biais.
- ⊙ Convergence rapide.
- ⊙ Sensible au bruit.

# LE BOOSTING, PREMIER ALGORITHME

Sommaire  
Aspects  
théoriques  
Mise en  
œuvre  
Applications  
Conclusion

## PREMIER MODÈLE



Source : Cours Machine Learning, Haytham ELGHAZEL

# LE BOOSTING, PREMIER ALGORITHME

Sommaire

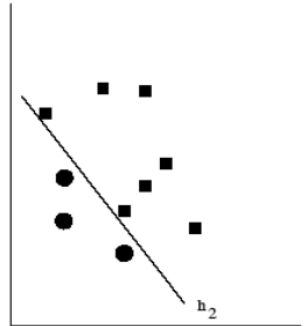
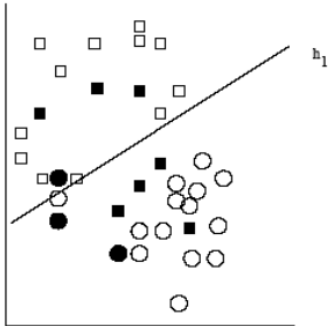
Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

## DEUXIÈME MODÈLE



Source : Cours Machine Learning, Haytham ELGHAZEL

# LE BOOSTING, PREMIER ALGORITHME

Sommaire

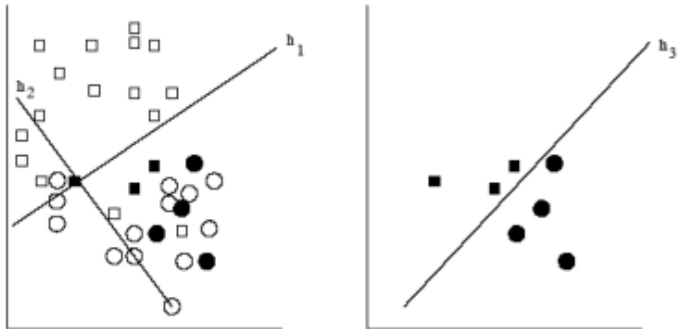
Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

## TROISIÈME MODÈLE



Source : Cours Machine Learning, Haytham ELGHAZEL



# LE BOOSTING, PREMIER ALGORITHME

Sommaire

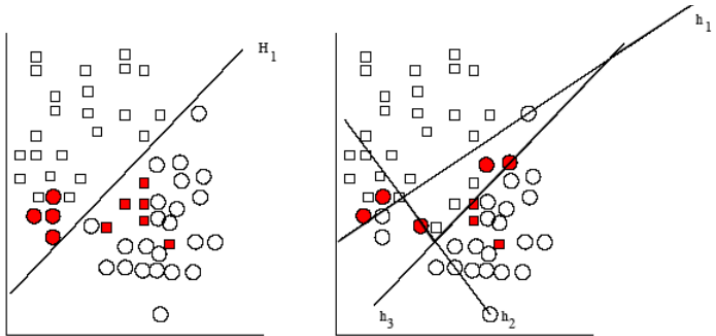
Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

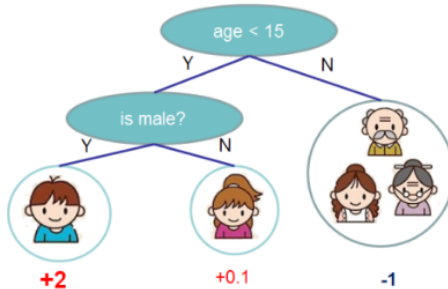
## VOTE MAJORITAIRE



Source : Cours Machine Learning, Haytham ELGHAZEL

## UN ARBRE SIMPLE (CART)

Does the person like computer games



Source : <https://xgboost.readthedocs.io/en/latest/model.html>

Sommaire

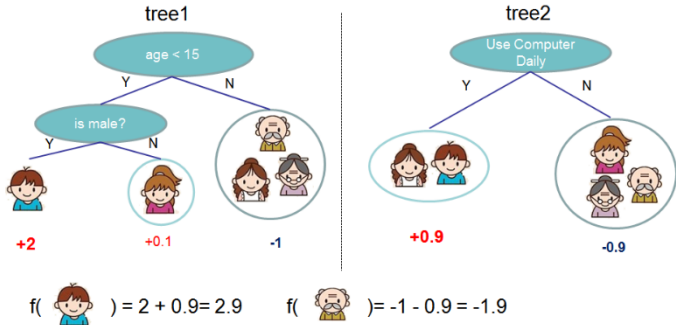
Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

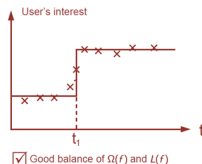
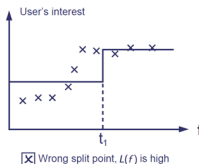
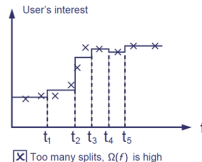
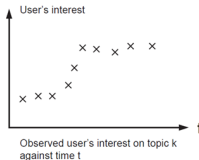
## PLUSIEURS ARBRES VALLENT MIEUX QU'UN



Source : <https://xgboost.readthedocs.io/en/latest/model.html>

## CHOIX DE L'ARBRE À AJOUTER

⊙ **Fonction objectif**  $\text{obj}(\theta) = L(\theta) + \Omega(\theta)$



Source : <https://xgboost.readthedocs.io/en/latest/model.html>





## CHOIX DE L'ARBRE À AJOUTER

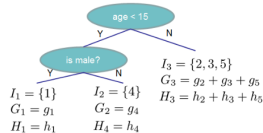
- Fonction de perte**

$$L(t) = \sum_{i=1}^n \left[ g_{if_t}(x_i) + \frac{1}{2} h_{if_t}^2(x_i) \right]$$
- Complexité**

$$\Omega(t) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T T \omega_j^2$$

Instance index    gradient statistics

1		$g_1, h_1$
2		$g_2, h_2$
3		$g_3, h_3$
4		$g_4, h_4$
5		$g_5, h_5$



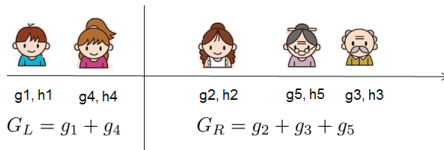
$$Obj = - \sum_j \frac{G_j^2}{H_j + \lambda} + 3\gamma$$

The smaller the score is, the better the structure is

Source : <https://xgboost.readthedocs.io/en/latest/model.html>

## CHOIX DE L'ARBRE À AJOUTER

⊙ **Le gain** 
$$\frac{1}{2} \left[ \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma$$



Source : <https://xgboost.readthedocs.io/en/latest/model.html>

Sommaire

Aspects  
théoriques

Mise en  
œuvre

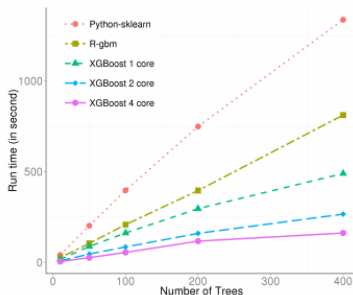
Applications

Conclusion

- ⊙ Prise en compte de la régularisation.
- ⊙ Calcul en parallèle.
- ⊙ Support de Hadoop.
- ⊙ Possibilité d'adaptation des fonctions objectifs.
- ⊙ Prise en charge des valeurs manquantes.
- ⊙ Version améliorée de l'élagage
- ⊙ Cross-validation native

La rapidité est le but initial de XGBoost :

- ⊙ **Mémoire** Pas de mémoire dynamique.
- ⊙ **Cache** Utilisation respectueuse.
- ⊙ **Amélioration modèle** Voir précédemment.
- ⊙ **Conception** Parallélisation en arrière plan.



Source : <http://www.jmlr.org/proceedings/papers/v42/chen14.pdf>



Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

## SUR LE BOOSTING

---

- ⊙ **1989** Boosting (R. Schapire)
- ⊙ **1996** AdaBoost (Y. Freund et R. Schapire)
- ⊙ **1999** GBM (L. Breiman puis J. Friedman)
- ⊙ **2014** XGBoost (T. Chen)

## POUR XGBOOST

---

- ⊙ **Mars 2014** Premières release
- ⊙ **Mai 2014** Python



Source : <http://homes.cs.washington.edu/Egchen/2016/03/10/story-and-lessons-behind-the-evolution-of-xgboost.html>

## POUR XGBOOST

- ⊙ **Mars 2014** Premières release
- ⊙ **Mai 2014** Python
- ⊙ **Septembre 2014** Parallélisation, R
- ⊙ **Mai 2015** YARN, gestion HDFS, SKLearn wrapper

scikit-learn  
gridsearch

R. caret  
grid search

scikit-learn  
classifier API

caret xgboost  
adaptor

XGBoost Python

XGBoost R

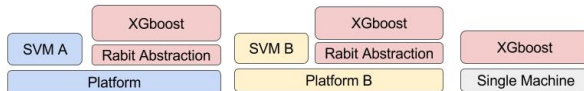
XGboost

XGboost

Source : <http://homes.cs.washington.edu/Egchen/2016/03/10/story-and-lessons-behind-the-evolution-of-xgboost.html>

## POUR XGBOOST

- ⊙ **Mars 2014** Premières release
- ⊙ **Mai 2014** Python
- ⊙ **Septembre 2014** Parallélisation, R
- ⊙ **Mai 2015** YARN, gestion HDFS, SKLearn wrapper
- ⊙ **Janvier 2016** API JAVA, amélioration R et Python
- ⊙ **Juillet 2016** C++11, JVM Package (JAVA et Scala)



Source : <http://homes.cs.washington.edu/eqchen/2016/03/10/story-and-lessons-behind-the-evolution-of-xgboost.html>

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

## PARAMÈTRES GÉNÉRIQUES

Pour définir par exemple quelle méthode Boosting sera utilisée.

## PARAMÈTRES LIÉS AU BOOSTING

Pour paramétrer le booster choisi.

## PARAMÈTRES LIÉS À L'APPRENTISSAGE

Dépend de la tâche d'apprentissage (classification,...).

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⊙ **Booster** Linéaire ou arbre.
- ⊙ **Silent** Affichage de messages.
- ⊙ **Nthread** Par défaut le maximum possible.

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

*Pour celui sur les arbres. Douze paramètres utiles...*

- ⊙ **eta**    Contrôle du niveau d'apprentissage.
- ⊙ **Min\_child\_weight**    Pour contrôler l'over/under-fitting
- ⊙ **Max\_depth**    Pour contrôler l'over-fitting.
- ⊙ **Subsample**    Fraction d'observations à utiliser pour les arbres.
- ⊙ **Lambda**    Pour de la régularisation.
- ⊙ ...



Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⊙ **Objective** Fonction objectif à minimiser (linéaire, softmax, softprob,...).
- ⊙ **Eval\_metric** Métrique d'évaluation (erreur MSE, MAE, LogLoss, AUC,...).
- ⊙ **Seed** Pour l'aléatoire.

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⊙ 1. Fixer un niveau d'apprentissage élevé
- ⊙ 2. Trouver le nombre optimal d'arbres
- ⊙ 3. Gérer les paramètres des arbres.
- ⊙ 4. Gérer les paramètres de régularisation.
- ⊙ 5. Réduire le niveau d'apprentissage.

# MÉDICAL : EXEMPLE DE LA GRIPPE

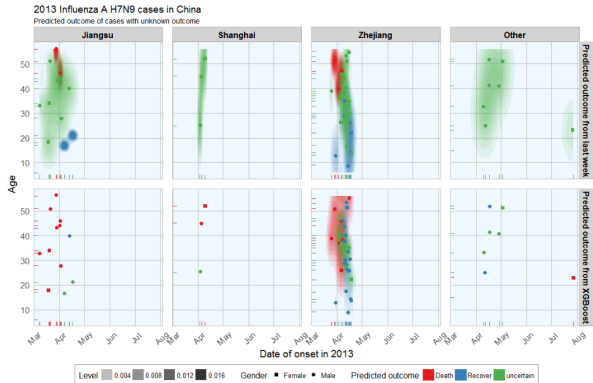
Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion



Source : [https://shiring.github.io/machine\\_learning/2016/12/02/flu\\_outcome\\_ML\\_2\\_post](https://shiring.github.io/machine_learning/2016/12/02/flu_outcome_ML_2_post)

Sommaire

Aspects  
théoriques

Mise en  
œuvre

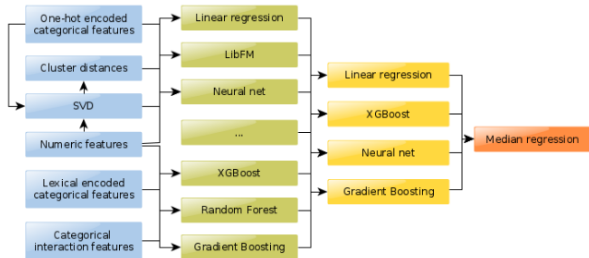
Applications

Conclusion

- ⊙ **1<sup>er</sup>** Knowledge Discovery and Data Mining Cup 2016 (V. Sandulescu).
- ⊙ **1<sup>er</sup> et 3<sup>ème</sup>** CERN LHCb experiment Flavour of Physics competition 2015 (V. Mironov).
- ⊙ **1<sup>er</sup>** Caterpillar Tube Pricing competition (M. Filho).
- ⊙ **2<sup>ème</sup>** Airbnb New User Bookings (K. Kuroyanagi).
- ⊙ **2<sup>ème</sup>** Allstate Claims Severity (A. Noskov).
- ⊙ **10%** Higgs Boson Competition (T. Chen).
- ⊙ ...

# UN EXEMPLE CONCRET

Sommaire  
Aspects  
théoriques  
Mise en  
œuvre  
Applications  
Conclusion



Source : <http://blog.kaggle.com/2017/02/27/allstate-claims-severity-competition-2nd-place-winners-interview-alezey-noskov/>

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

*Des données difficiles à obtenir...*

- ⦿ ODPS Cloud Service (Alibaba)
- ⦿ Tencent (QQ)
- ⦿ AutoHome
- ⦿ AXA, Expedia, Amazon,...

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⊙ Une implémentation récente (2 ans).
- ⊙ Une forte portabilité et de bonnes performances.
- ⊙ Une utilisation industrielle qui semble se développer.
- ⊙ Savoir-faire nécessaire pour la configuration

Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

- ⊙ Une implémentation récente (2 ans).
- ⊙ Une forte portabilité et de bonnes performances.
- ⊙ Une utilisation industrielle qui semble se développer.
- ⊙ Savoir-faire nécessaire pour la configuration

**Une solution qui semble avoir de l'avenir!**



Sommaire

Aspects  
théoriques

Mise en  
œuvre

Applications

Conclusion

**Merci pour votre attention  
Et place aux questions!**