

Compléments pour l'Apprentissage Statistique

Cycle Ingénieur - 3^{eme} année

Paul CHAPRON, Juste RAIMBAULT, Yann MÉNEROUX

paul.chapron@ign.fr
juste.raimbault@ign.fr
yann.meneroux@ign.fr

décembre 2022

Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Indicateurs de performance

On se place dans le cadre de l'application d'un classifieur binaire sur un jeu de données de validation contenant $N = TN + FP$ exemples négatifs et $P = TP + FN$ exemples positifs.

Y/\hat{Y}	0	1
0	TN	FP
1	FN	TP

On notera $P^+ = P/(P + N)$ et $P^- = 1 - P^+ = N/(P + N)$.

Indicateurs de performance

Précisions *producteur*

Sensibilité STV

C'est le pourcentage d'exemples positifs détectés :

$$\mathbb{P}[\hat{Y} = 1 \mid Y = 1] = \frac{TP}{TP + FN}$$

où Y et \hat{Y} désignent respectivement la variable cible *réelle* et *estimée*. On l'appelle aussi **rappel** ou **TPR** (True Positive Rate).

Spécificité SPC

C'est le pourcentage d'exemples négatifs (à raison) non-détectés :

$$\mathbb{P}[\hat{Y} = 0 \mid Y = 0] = \frac{TN}{TN + FP}$$

On a : $SPC = 1 - FPR$, où $FPR = \text{False Positive Rate}$.

Indicateurs de performance

Précisions *utilisateur*

Précision PPV

C'est le pourcentage d'exemples détectés positifs corrects :

$$\mathbb{P}[Y = 1 \mid \hat{Y} = 1] = \frac{TP}{TP + FP}$$

Précision NPV

C'est le pourcentage d'exemples détectés négatifs corrects :

$$\mathbb{P}[Y = 0 \mid \hat{Y} = 0] = \frac{TN}{TN + FN}$$

Contrairement aux précisions *producteur*, les précisions *utilisateur* dépendent de la proportion (positifs/négatifs) du jeu de données sur lequel elles sont évaluées.

Précisions producteur \leftrightarrow utilisateur

On peut passer des précisions **producteur** aux précisions **utilisateur** (et réciproquement) à l'aide du théorème de Bayes, et à condition de connaître les **proportions** du jeu de données. Par exemple :

Précisions producteur ↔ utilisateur

On peut passer des précisions **producteur** aux précisions **utilisateur** (et réciproquement) à l'aide du théorème de Bayes, et à condition de connaître les **proportions** du jeu de données. Par exemple :

$$PPV = \mathbb{P}[Y = 1 \mid \hat{Y} = 1] = \frac{\mathbb{P}[\hat{Y} = 1 \mid Y = 1]\mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]}$$

Précisions producteur ↔ utilisateur

On peut passer des précisions **producteur** aux précisions **utilisateur** (et réciproquement) à l'aide du théorème de Bayes, et à condition de connaître les **proportions** du jeu de données. Par exemple :

$$\begin{aligned} \textcolor{red}{PPV} &= \mathbb{P}[Y = 1 \mid \hat{Y} = 1] = \frac{\mathbb{P}[\hat{Y} = 1 \mid Y = 1]\mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]} \\ &= \frac{\textcolor{blue}{STV} \times \mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]} \end{aligned}$$

Précisions producteur ↔ utilisateur

On peut passer des précisions **producteur** aux précisions **utilisateur** (et réciproquement) à l'aide du théorème de Bayes, et à condition de connaître les **proportions** du jeu de données. Par exemple :

$$PPV = \mathbb{P}[Y = 1 \mid \hat{Y} = 1] = \frac{\mathbb{P}[\hat{Y} = 1 \mid Y = 1]\mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]}$$

$$= \frac{STV \times \mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]}$$

$$= \frac{STV \times P^+}{\sum_y \mathbb{P}[\hat{Y} = 1, Y = y]} = \frac{STV \times P^+}{\sum_y \mathbb{P}[\hat{Y} = 1 | Y = y]\mathbb{P}[Y = y]}$$

Précisions producteur ↔ utilisateur

On peut passer des précisions **producteur** aux précisions **utilisateur** (et réciproquement) à l'aide du théorème de Bayes, et à condition de connaître les **proportions** du jeu de données. Par exemple :

$$PPV = \mathbb{P}[Y = 1 \mid \hat{Y} = 1] = \frac{\mathbb{P}[\hat{Y} = 1 \mid Y = 1]\mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]}$$

$$= \frac{STV \times \mathbb{P}[Y = 1]}{\mathbb{P}[\hat{Y} = 1]}$$

$$= \frac{STV \times P^+}{\sum_y \mathbb{P}[\hat{Y} = 1, Y = y]} = \frac{STV \times P^+}{\sum_y \mathbb{P}[\hat{Y} = 1 \mid Y = y]\mathbb{P}[Y = y]}$$

$$= \frac{STV \times P^+}{(1 - SPC) \times P^- + STV \times P^+}$$

Overall accuracy

C'est le pourcentage d'exemples correctement détectés :

$$ACC = \mathbb{P}[\hat{Y} = Y] = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\begin{aligned} ACC &= \mathbb{P}[\hat{Y} = 0, Y = 0] + \mathbb{P}[\hat{Y} = 1, Y = 1] \\ &= \mathbb{P}[\hat{Y} = 0|Y = 0] \times P^- + \mathbb{P}[\hat{Y} = 1|Y = 1] \times P^+ \\ &= SPC \times P^- + STV \times P^+ \end{aligned}$$

Mesure F_1

C'est la moyenne harmonique de la précision (PPV) et du rappel (STV) :

$$F_1 = \left(\frac{PPV^{-1} + STV^{-1}}{2} \right)^{-1}$$

On peut la généraliser en mesure $(F_\beta)_{\beta \in \mathbb{N}^*}$

Puisqu'elle dépend du rappel, la mesure F_1 dépend aussi des proportions du jeu de données sur lequel on l'évalue. Elle permet de mesurer la complétude et l'exactitude de la détection.

Indicateurs agrégés

En notant p la précision et r le rappel, on a :

$$F_1(p, r) = \left(\frac{p^{-1} + r^{-1}}{2} \right)^{-1} = \frac{2pr}{p+r}$$

Indicateurs agrégés

En notant p la précision et r le rappel, on a :

$$F_1(p, r) = \left(\frac{p^{-1} + r^{-1}}{2} \right)^{-1} = \frac{2pr}{p+r}$$

$$\frac{\partial F_1}{\partial p}(p, r) = \frac{2r(p+r) - 2pr}{(p+r)^2} = \frac{2r^2}{(p+r)^2}$$

Indicateurs agrégés

En notant p la précision et r le rappel, on a :

$$F_1(p, r) = \left(\frac{p^{-1} + r^{-1}}{2} \right)^{-1} = \frac{2pr}{p+r}$$

$$\frac{\partial F_1}{\partial p}(p, r) = \frac{2r(p+r) - 2pr}{(p+r)^2} = \frac{2r^2}{(p+r)^2}$$

$$\frac{\partial F_1}{\partial p}(p, r) = \begin{cases} 0 & \text{si } p \gg r \\ 2 & \text{si } r \gg p \end{cases}$$

Indicateurs agrégés

En notant p la précision et r le rappel, on a :

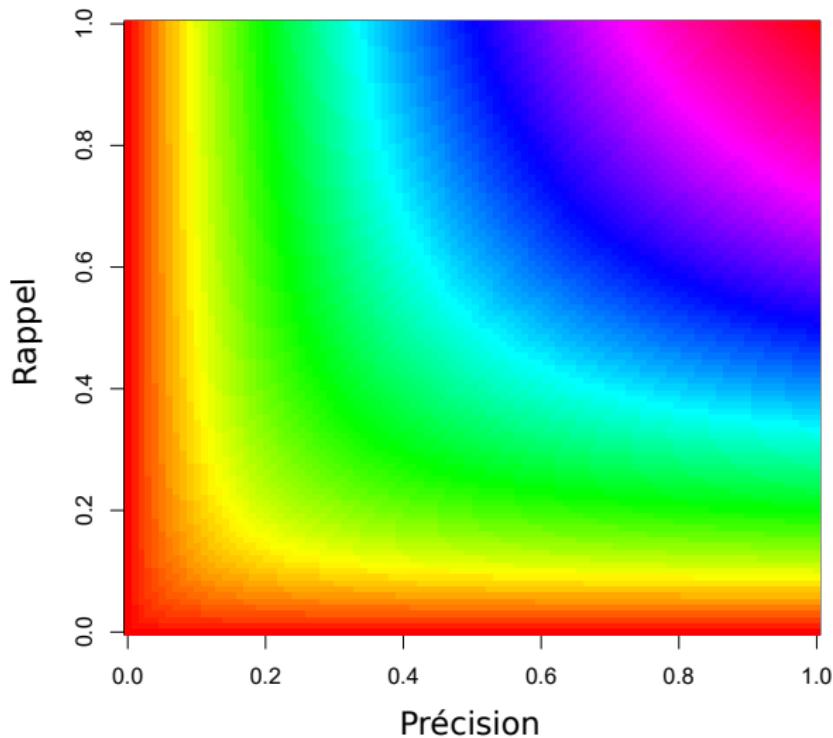
$$F_1(p, r) = \left(\frac{p^{-1} + r^{-1}}{2} \right)^{-1} = \frac{2pr}{p+r}$$

$$\frac{\partial F_1}{\partial p}(p, r) = \frac{2r(p+r) - 2pr}{(p+r)^2} = \frac{2r^2}{(p+r)^2}$$

$$\frac{\partial F_1}{\partial p}(p, r) = \begin{cases} 0 & \text{si } p \gg r \\ 2 & \text{si } r \gg p \end{cases}$$

La mesure F_1 pénalise fortement l'indicateur le plus faible !

Indicateurs agrégés



La mesure F_1 pénalise fortement l'indicateur le plus faible !

On considère un classifieur binaire dont les précisions *producteur* ont été évaluées "en laboratoire" à 75% de sensibilité (STV) et 95% de spécificité (SPC).

Calculer la mesure F_1 obtenue lorsqu'on applique ce classifieur sur un jeu de données équilibré (50% d'instances positives/négatives).

Exercice

On considère un classifieur binaire dont les précisions *producteur* ont été évaluées "en laboratoire" à 75% de sensibilité (STV) et 95% de spécificité (SPC).

Calculer la mesure F_1 obtenue lorsqu'on applique ce classifieur sur un jeu de données équilibré (50% d'instances positives/négatives).

Réponse :

$$\text{Rappel} = \text{STV} = 0.75$$

$$\text{Precision} = \text{PPV} = 0.9375$$

$$F_1 = 83.33 \%$$

Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Incertitude de l'évaluation

Bien souvent, le jeu de données de validation est limité, et les indicateurs présentés précédemment sont entachés d'une erreur qu'il est important de quantifier et maîtriser.

Incertitude de l'évaluation

Bien souvent, le jeu de données de validation est limité, et les indicateurs présentés précédemment sont entachés d'une erreur qu'il est important de quantifier et maîtriser.

Pour les indicateurs simples *proportionnels* (PPV, SPC, STV...), on utilise les règles classiques vues en cours de Probas & Stats : $M = (X_1 + X_2 + \dots + X_n)/n$ sous l'hypothèse de données *i.i.d.* :

Incertitude de l'évaluation

Bien souvent, le jeu de données de validation est limité, et les indicateurs présentés précédemment sont entachés d'une erreur qu'il est important de quantifier et maîtriser.

Pour les indicateurs simples *proportionnels* (PPV, SPC, STV...), on utilise les règles classiques vues en cours de Probas & Stats : $M = (X_1 + X_2 + \dots + X_n)/n$ sous l'hypothèse de données *i.i.d.* :

$$\mathbb{V}[M] = \frac{1}{n^2} \mathbb{V}\left[\sum_{i=1}^n X_i\right] = \frac{1}{n^2} \sum_{i=1}^n \mathbb{V}[X_i] = \frac{1}{n} \mathbb{V}[X]$$

Incertitude de l'évaluation

Bien souvent, le jeu de données de validation est limité, et les indicateurs présentés précédemment sont entachés d'une erreur qu'il est important de quantifier et maîtriser.

Pour les indicateurs simples *proportionnels* (PPV, SPC, STV...), on utilise les règles classiques vues en cours de Probas & Stats : $M = (X_1 + X_2 + \dots + X_n)/n$ sous l'hypothèse de données *i.i.d.* :

$$\mathbb{V}[M] = \frac{1}{n^2} \mathbb{V}\left[\sum_{i=1}^n X_i\right] = \frac{1}{n^2} \sum_{i=1}^n \mathbb{V}[X_i] = \frac{1}{n} \mathbb{V}[X]$$

$$\sigma_M = \frac{\sigma_X}{\sqrt{n}}$$

Incertitude de l'évaluation

Bien souvent, le jeu de données de validation est limité, et les indicateurs présentés précédemment sont entâchés d'une errreur qu'il est important de quantifier et maîtriser.

Pour les indicateurs simples *proportionnels* (PPV, SPC, STV...), on utilise les règles classiques vues en cours de Probas & Stats : $M = (X_1 + X_2 + \dots + X_n)/n$ sous l'hypothèse de données *i.i.d.* :

$$\begin{aligned}\mathbb{V}[M] &= \frac{1}{n^2} \mathbb{V}\left[\sum_{i=1}^n X_i\right] = \frac{1}{n^2} \sum_{i=1}^n \mathbb{V}[X_i] = \frac{1}{n} \mathbb{V}[X] \\ \sigma_M &= \frac{\sigma_X}{\sqrt{n}}\end{aligned}$$

Une "règle du pouce" : pour un jeu de validation contenant N exemples négatifs et P exemples positifs, on peut considérer que l'incertitude entâchant les indicateurs est de l'ordre de $1/\sqrt{\min(N, P)}$.

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de...

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de... $1/\sqrt{57} = 13\%$

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de... $1/\sqrt{57} = 13\%$

⇒ aucune conclusion possible sur la comparaison de deux modèles dont les performances sont indiscernables à 13 % près.

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de... $1/\sqrt{57} = 13\%$

⇒ aucune conclusion possible sur la comparaison de deux modèles dont les performances sont indiscernables à 13 % près.

Pour des indicateurs I plus complexes (mesure F_1 , etc), on utilise la règle de propagation des variances :

Incertitude de l'évaluation

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de... $1/\sqrt{57} = 13\%$

⇒ aucune conclusion possible sur la comparaison de deux modèles dont les performances sont indiscernables à 13 % près.

Pour des indicateurs I plus complexes (mesure F_1 , etc), on utilise la règle de propagation des variances :

$$\Sigma_I = \mathbf{A}\Sigma_X\mathbf{A}^T \quad \text{ou} \quad \Sigma_I = \mathbf{J}\Sigma_X\mathbf{J}^T \quad (\text{cas non-linéaire})$$

où \mathbf{J} est la jacobienne de la fonction de calcul de l'indicateur.

Incertitude de l'évaluation

Exemple : un jeu de validation contient 213 exemples positifs et 57 exemples négatifs. L'incertitude typique (confiance à 95 %) d'un indicateur de précision sera de l'ordre de... $1/\sqrt{57} = 13\%$

⇒ aucune conclusion possible sur la comparaison de deux modèles dont les performances sont indiscernables à 13 % près.

Pour des indicateurs I plus complexes (mesure F_1 , etc), on utilise la règle de propagation des variances :

$$\Sigma_I = \mathbf{A}\Sigma_X\mathbf{A}^T \quad \text{ou} \quad \Sigma_I = \mathbf{J}\Sigma_X\mathbf{J}^T \quad (\text{cas non-linéaire})$$

où \mathbf{J} est la jacobienne de la fonction de calcul de l'indicateur.

Et si ça ne suffit toujours pas... ?

L'artillerie lourde de la Statistique...

L'artillerie lourde de la Statistique...

On considère un indicateur $Y \in \mathbb{R}$, calculé par une fonctionnelle F , prenant en inputs un vecteur de données $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$:

$$\begin{aligned} F &: \mathbb{R}^n \mapsto \mathbb{R} \\ \mathbf{x} &\mapsto Y = F(\mathbf{x}) \end{aligned}$$

L'artillerie lourde de la Statistique...

On considère un indicateur $Y \in \mathbb{R}$, calculé par une fonctionnelle F , prenant en inputs un vecteur de données $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$:

$$\begin{aligned} F &: \mathbb{R}^n \mapsto \mathbb{R} \\ \mathbf{x} &\mapsto Y = F(\mathbf{x}) \end{aligned}$$

Recette : on génère une série de N échantillons bootstrap $B = (\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_N)$, où \mathbf{b}_i est un réechantillonnage avec remise de \mathbf{x} .

L'artillerie lourde de la Statistique...

On considère un indicateur $Y \in \mathbb{R}$, calculé par une fonctionnelle F , prenant en inputs un vecteur de données $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$:

$$\begin{aligned} F &: \mathbb{R}^n \mapsto \mathbb{R} \\ \mathbf{x} &\mapsto Y = F(\mathbf{x}) \end{aligned}$$

Recette : on génère une série de N échantillons bootstrap $B = (\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_N)$, où \mathbf{b}_i est un réechantillonnage avec remise de \mathbf{x} .

On calcule $Y_i = F(\mathbf{b}_i)$ pour chaque échantillon bootstrap \mathbf{b}_i .

L'artillerie lourde de la Statistique...

On considère un indicateur $Y \in \mathbb{R}$, calculé par une fonctionnelle F , prenant en inputs un vecteur de données $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$:

$$\begin{aligned} F &: \mathbb{R}^n \mapsto \mathbb{R} \\ \mathbf{x} &\mapsto Y = F(\mathbf{x}) \end{aligned}$$

Recette : on génère une série de N échantillons bootstrap $B = (\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_N)$, où \mathbf{b}_i est un réechantillonnage avec remise de \mathbf{x} .

On calcule $Y_i = F(\mathbf{b}_i)$ pour chaque échantillon bootstrap \mathbf{b}_i .

On extrait alors toutes les statistiques souhaitées sur la v.a. Y à partir d'estimateurs empiriques sur ses réalisations bootstrap Y_i .

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$$

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$

`x[sample(1:5, 5, replace=TRUE)]` :

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$$

`x[sample(1:5, 5, replace=TRUE)] :`

$$\mathbf{b}_1 = (15.122, 15.126, 15.126, 15.126, 15.122) \quad F(\mathbf{b}_1) = 15.1244$$

Bootstrap

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$$

`x [sample(1:5, 5, replace=TRUE)] :`

$$\mathbf{b}_1 = (15.122, 15.126, 15.126, 15.126, 15.122) \quad F(\mathbf{b}_1) = 15.1244$$

$$\mathbf{b}_2 = (15.126, 15.121, 15.122, 15.124, 15.122) \quad F(\mathbf{b}_2) = 15.1230$$

$$\mathbf{b}_3 = (15.121, 15.121, 15.126, 15.121, 15.121) \quad F(\mathbf{b}_3) = 15.1220$$

$$\mathbf{b}_4 = (15.126, 15.121, 15.124, 15.126, 15.122) \quad F(\mathbf{b}_4) = 15.1238$$

Bootstrap

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$$

`x[sample(1:5, 5, replace=TRUE)]` :

$$\mathbf{b}_1 = (15.122, 15.126, 15.126, 15.126, 15.122) \quad F(\mathbf{b}_1) = 15.1244$$

$$\mathbf{b}_2 = (15.126, 15.121, 15.122, 15.124, 15.122) \quad F(\mathbf{b}_2) = 15.1230$$

$$\mathbf{b}_3 = (15.121, 15.121, 15.126, 15.121, 15.121) \quad F(\mathbf{b}_3) = 15.1220$$

$$\mathbf{b}_4 = (15.126, 15.121, 15.124, 15.126, 15.122) \quad F(\mathbf{b}_4) = 15.1238$$

$$\mu = 15.122 \pm 1 \text{ mm}$$

Exemple : évaluation de l'incertitude sur 5 mesures de distance :

$$\mathbf{x} = (15.124, 15.121, 15.122, 15.121, 15.126) \quad \mu = F(\mathbf{x}) = 15.122 \text{ m}$$

`x[sample(1:5, 5, replace=TRUE)]` :

$$\mathbf{b}_1 = (15.122, 15.126, 15.126, 15.126, 15.122) \quad F(\mathbf{b}_1) = 15.1244$$

$$\mathbf{b}_2 = (15.126, 15.121, 15.122, 15.124, 15.122) \quad F(\mathbf{b}_2) = 15.1230$$

$$\mathbf{b}_3 = (15.121, 15.121, 15.126, 15.121, 15.121) \quad F(\mathbf{b}_3) = 15.1220$$

$$\mathbf{b}_4 = (15.126, 15.121, 15.124, 15.126, 15.122) \quad F(\mathbf{b}_4) = 15.1238$$

$$\mu = 15.122 \pm 1 \text{ mm}$$

En pratique $N \gg 4$ (100, 200, 1000...)

Courbe ROC

Un classifieur binaire utilisé en mode *probabiliste* retourne, pour chaque nouvelle instance x , la probabilité : $p = \mathbb{P}(Y = 1 \mid x)$, la décision finale étant calculée par seuillage par l'utilisateur.

Courbe ROC

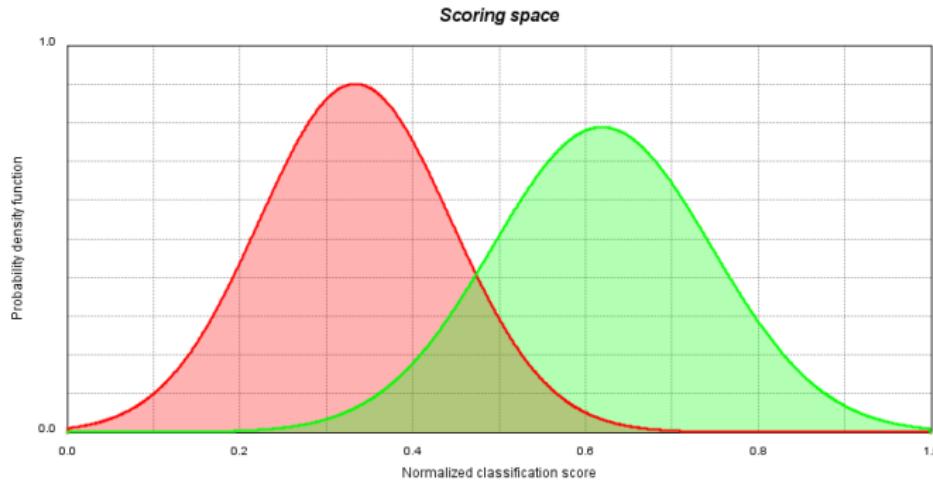
Un classifieur binaire utilisé en mode *probabiliste* retourne, pour chaque nouvelle instance \mathbf{x} , la probabilité : $p = \mathbb{P}(Y = 1 \mid \mathbf{x})$, la décision finale étant calculée par seuillage par l'utilisateur.

Il peut être instructif de regarder les distributions des valeurs p pour chaque classe (+/-) :

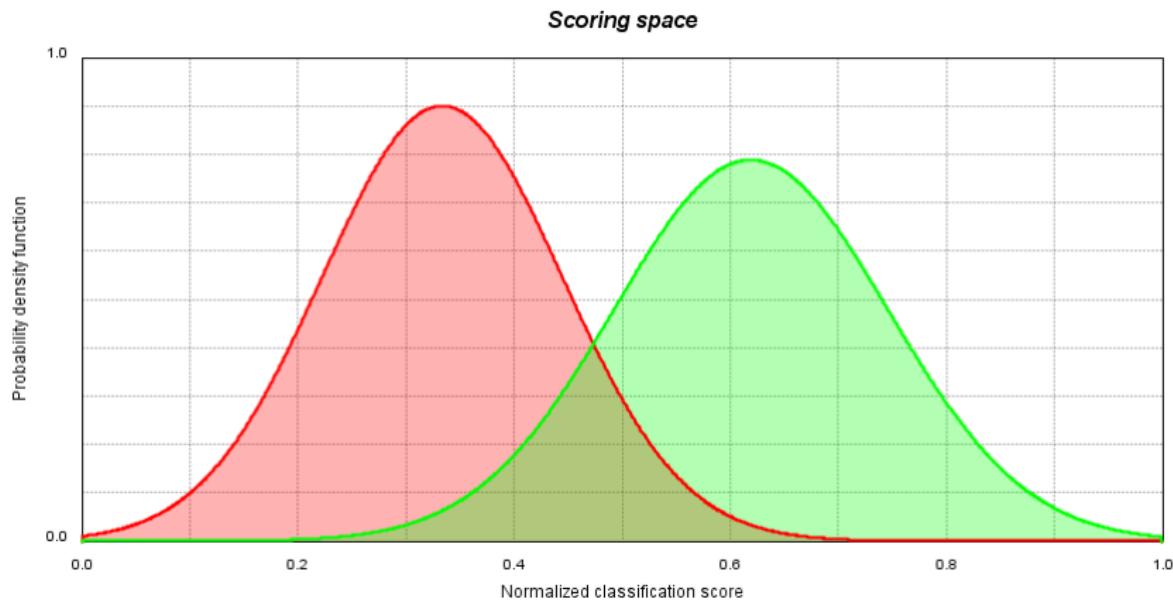
Courbe ROC

Un classifieur binaire utilisé en mode *probabiliste* retourne, pour chaque nouvelle instance x , la probabilité : $p = \mathbb{P}(Y = 1 | x)$, la décision finale étant calculée par seuillage par l'utilisateur.

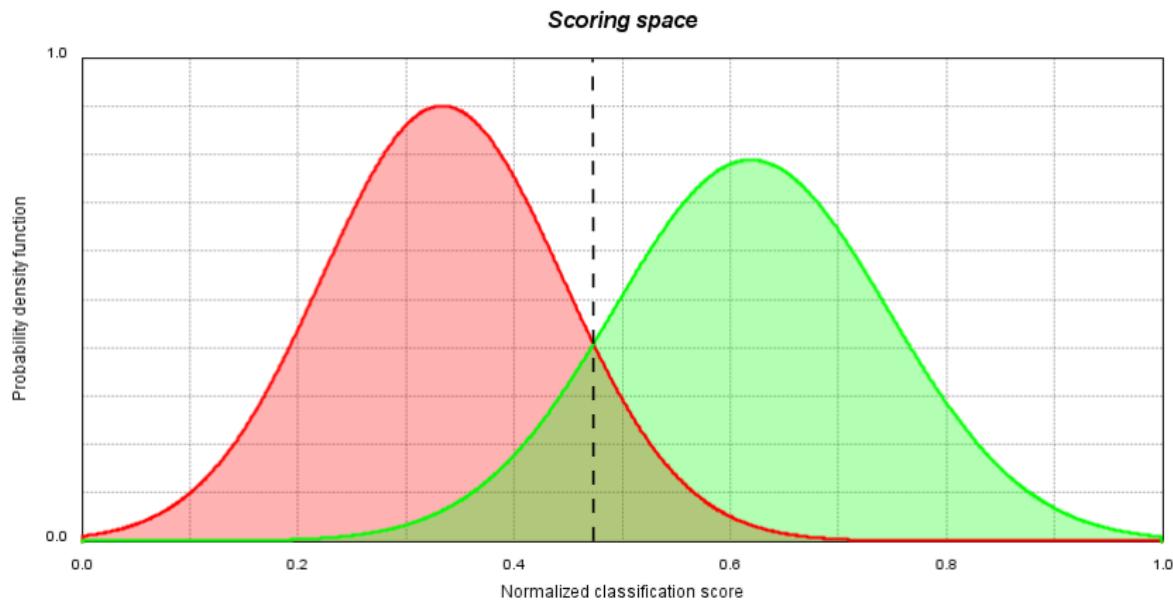
Il peut être instructif de regarder les distributions des valeurs p pour chaque classe (+/-) :



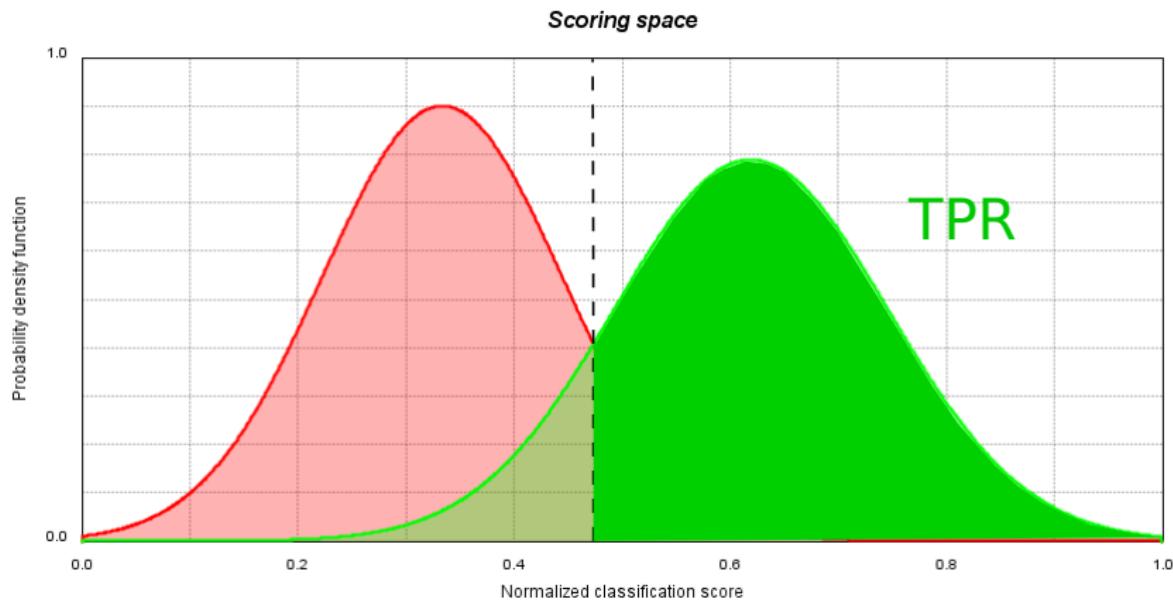
Courbe ROC



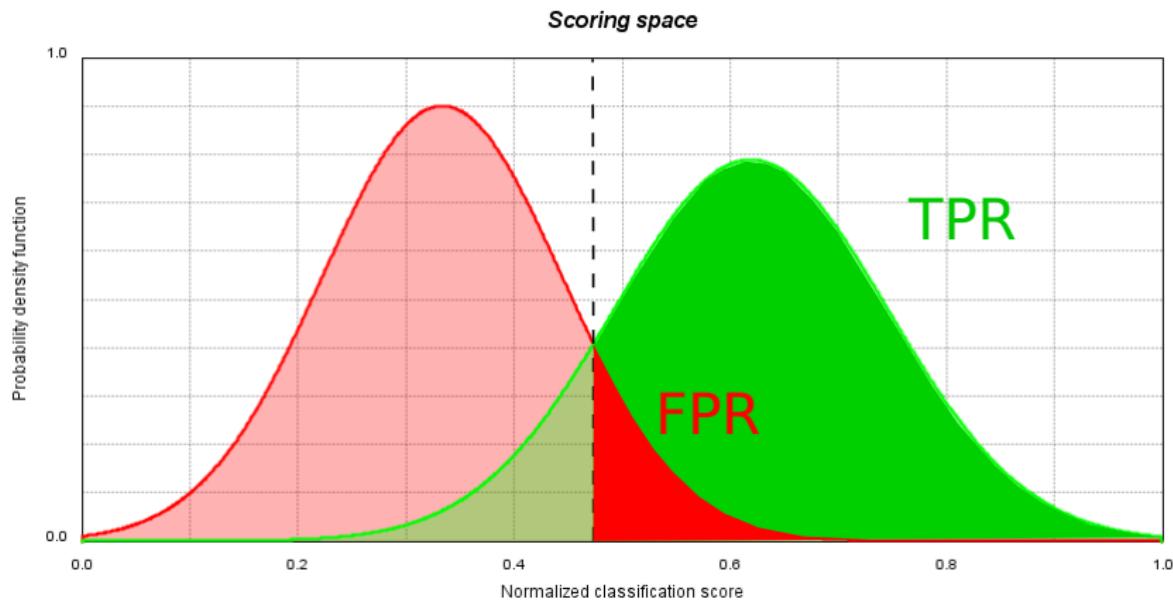
Courbe ROC



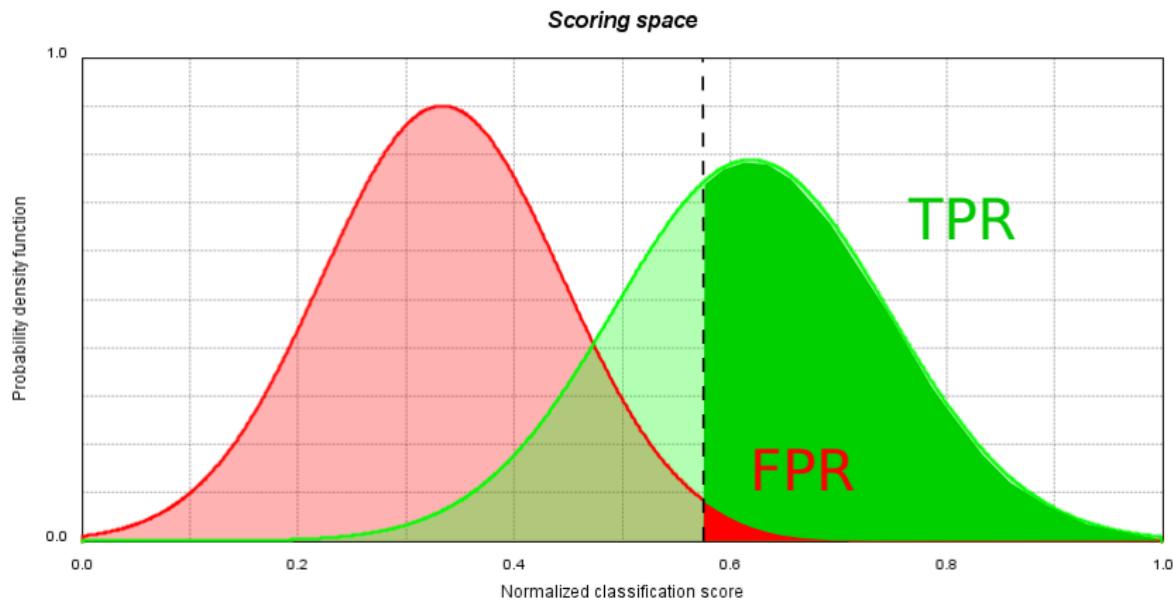
Courbe ROC



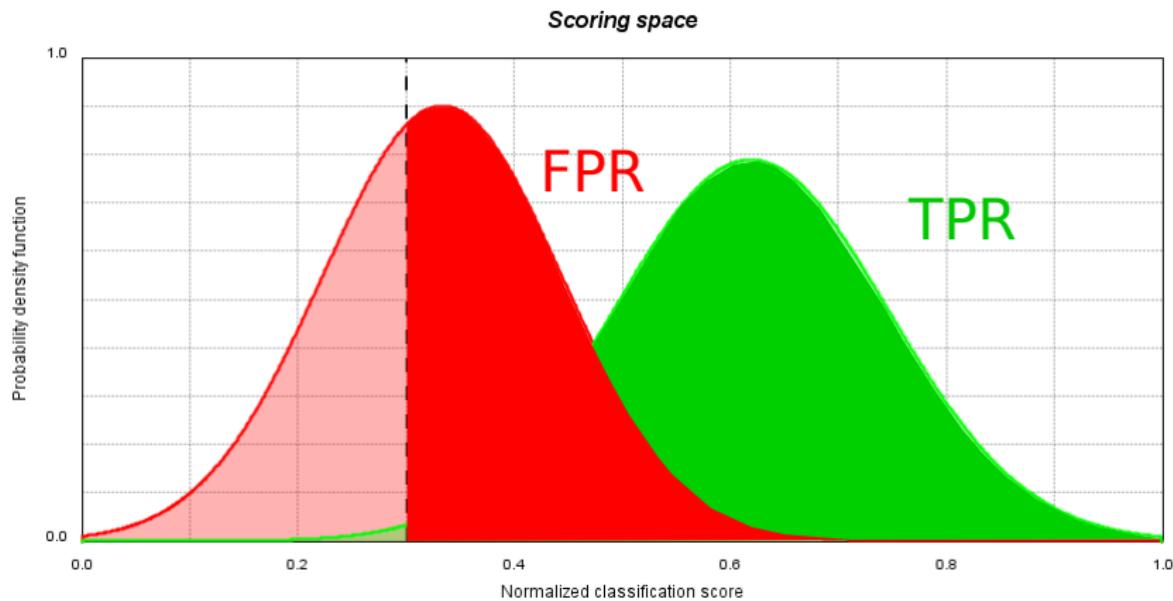
Courbe ROC



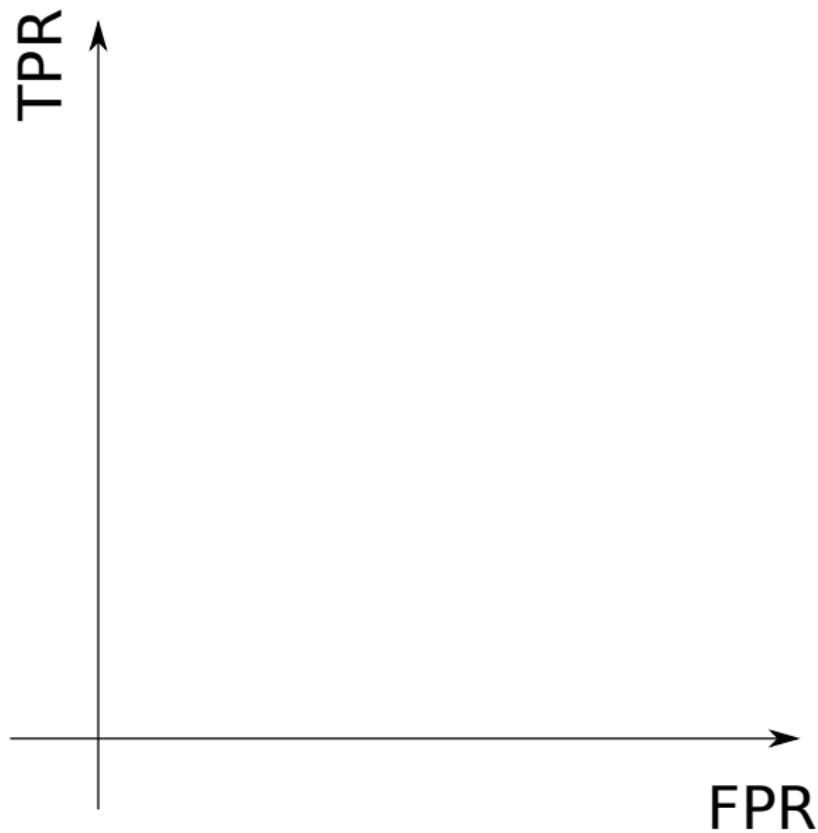
Courbe ROC



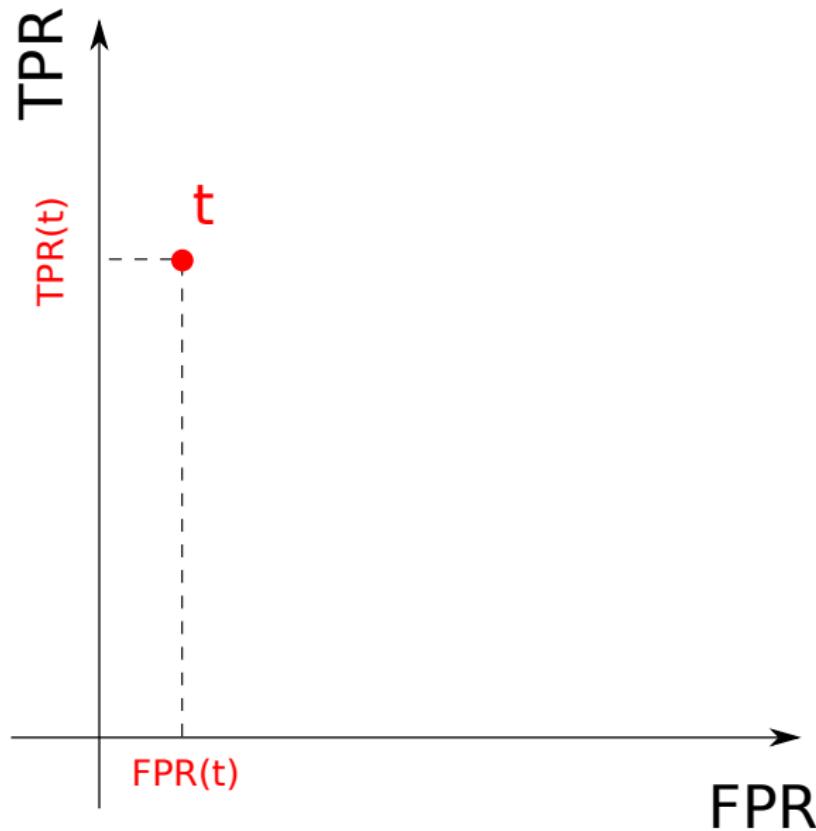
Courbe ROC



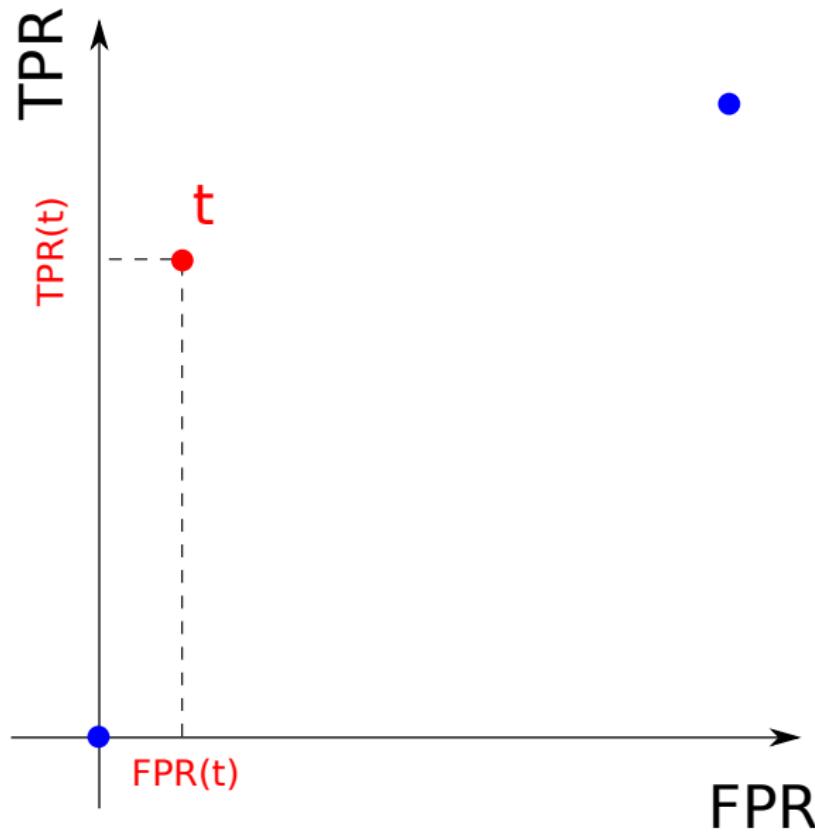
Courbe ROC



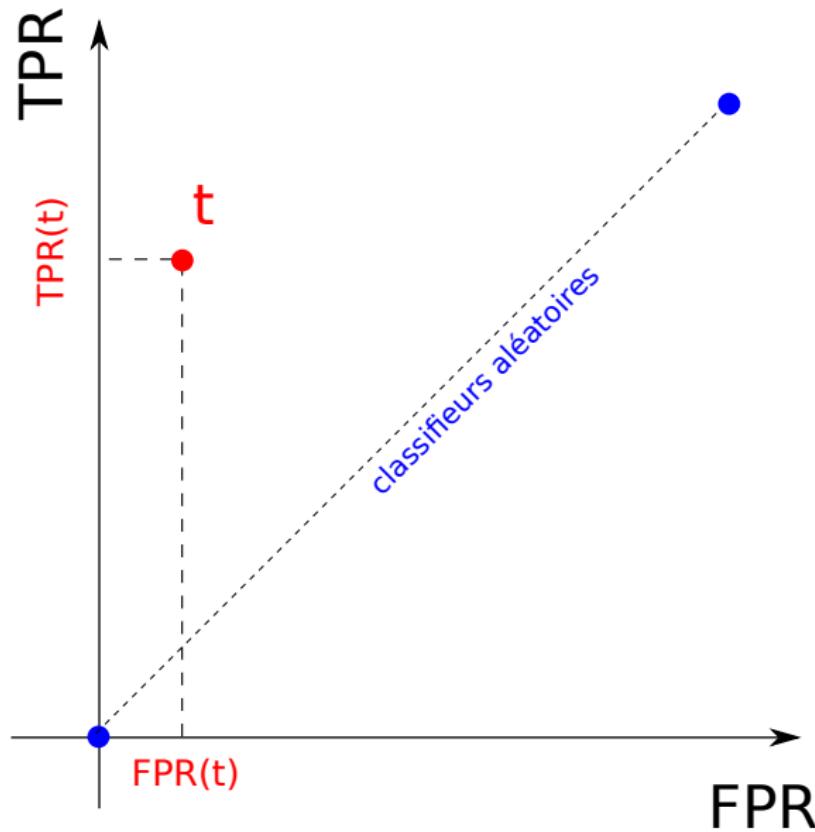
Courbe ROC



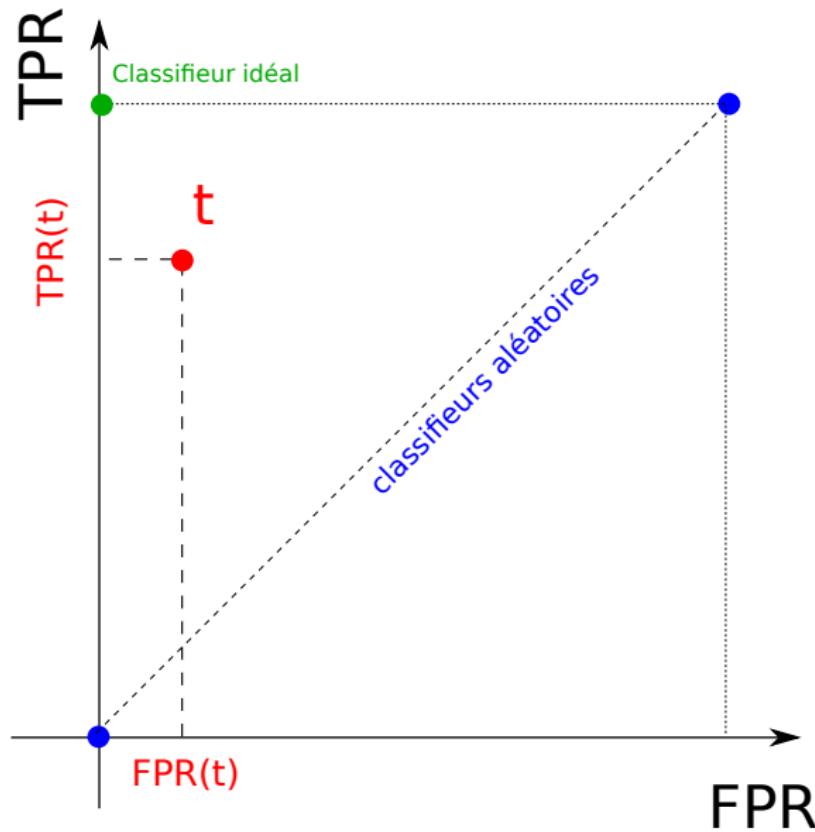
Courbe ROC



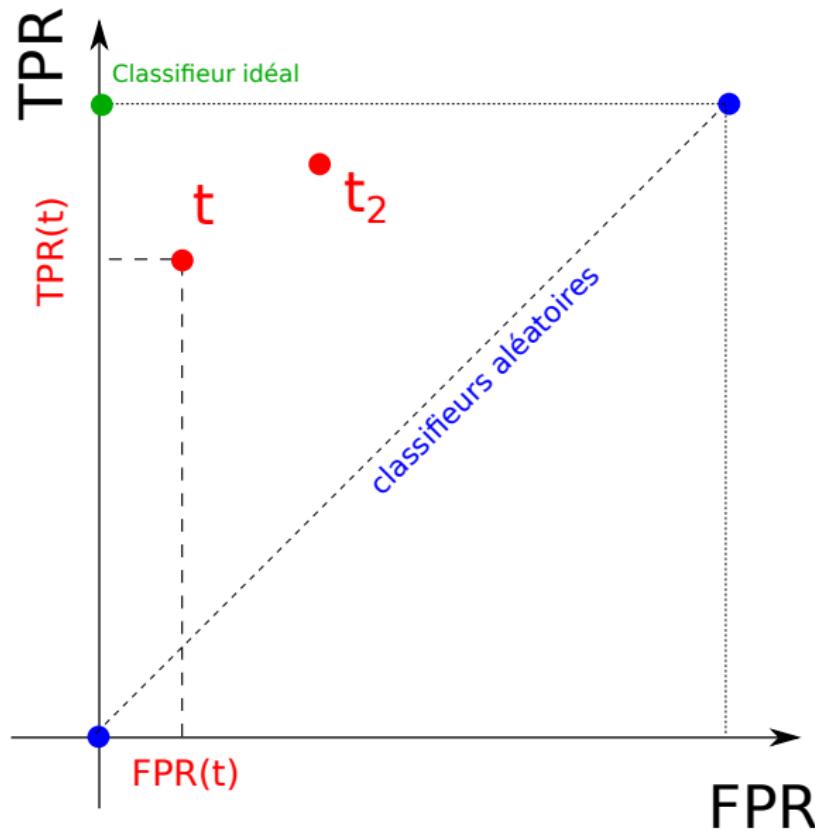
Courbe ROC



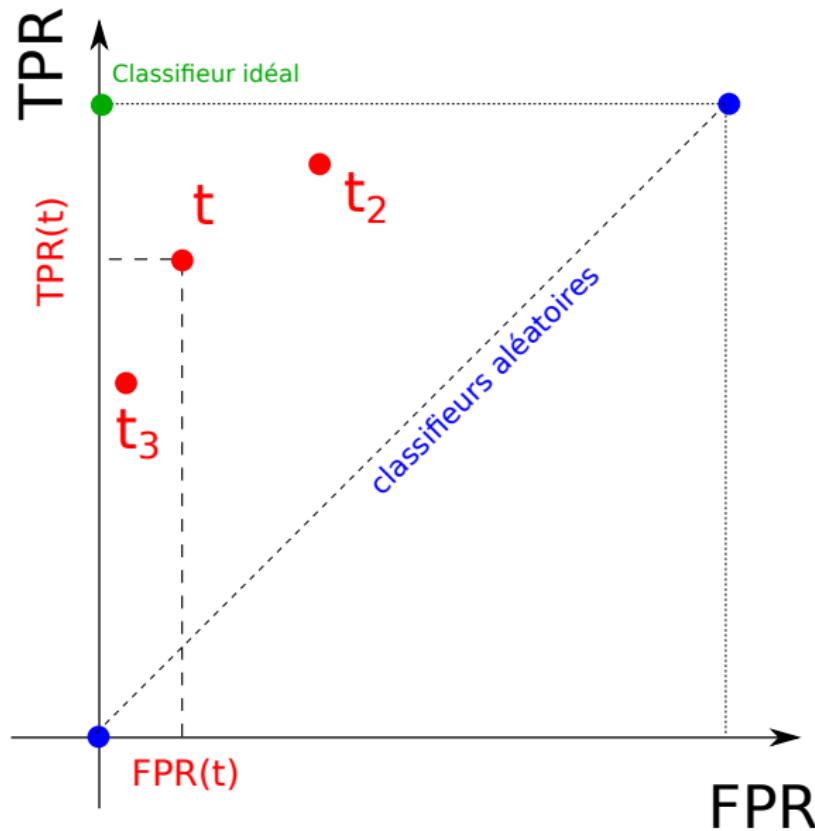
Courbe ROC



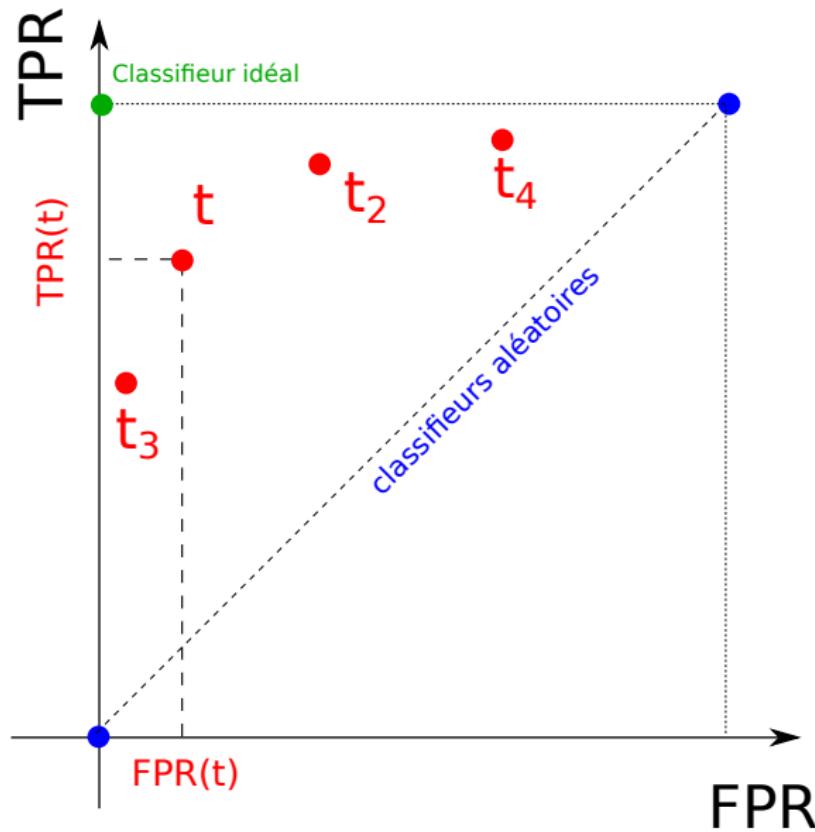
Courbe ROC



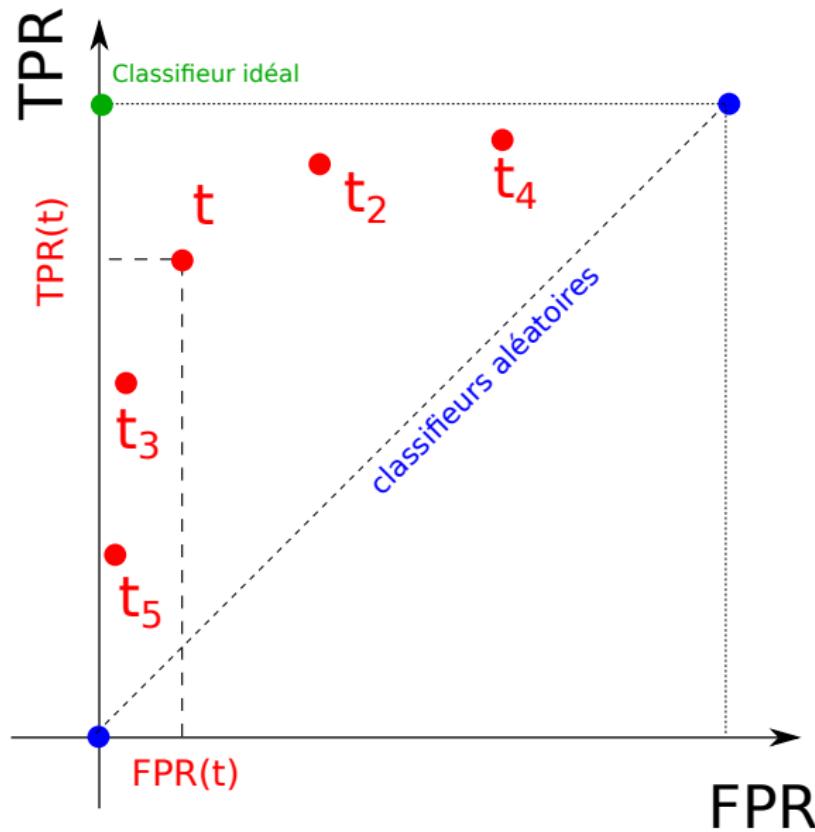
Courbe ROC



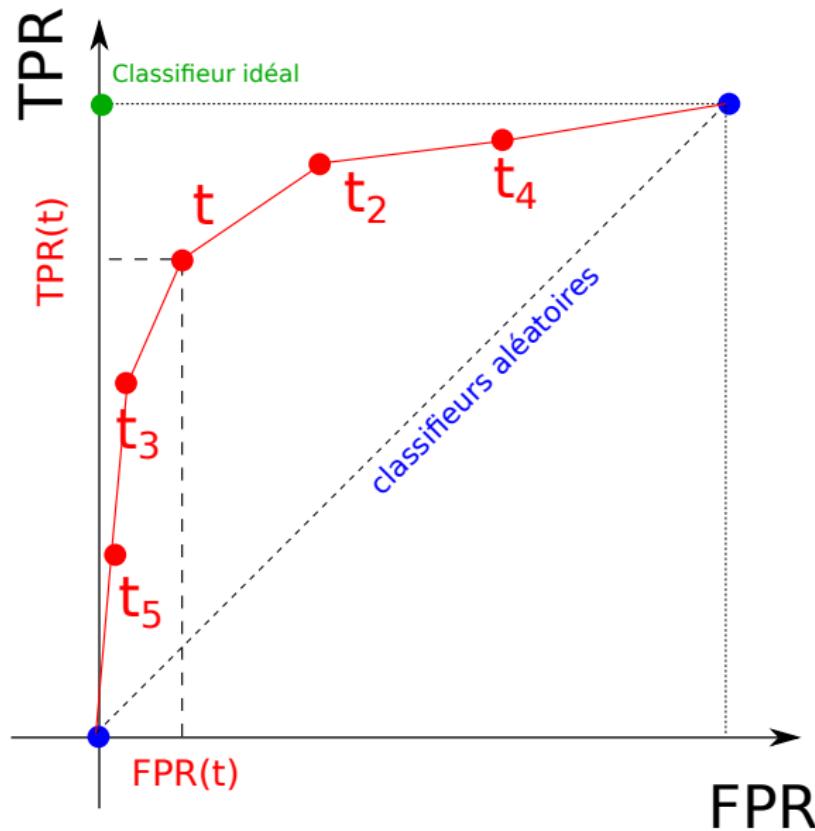
Courbe ROC



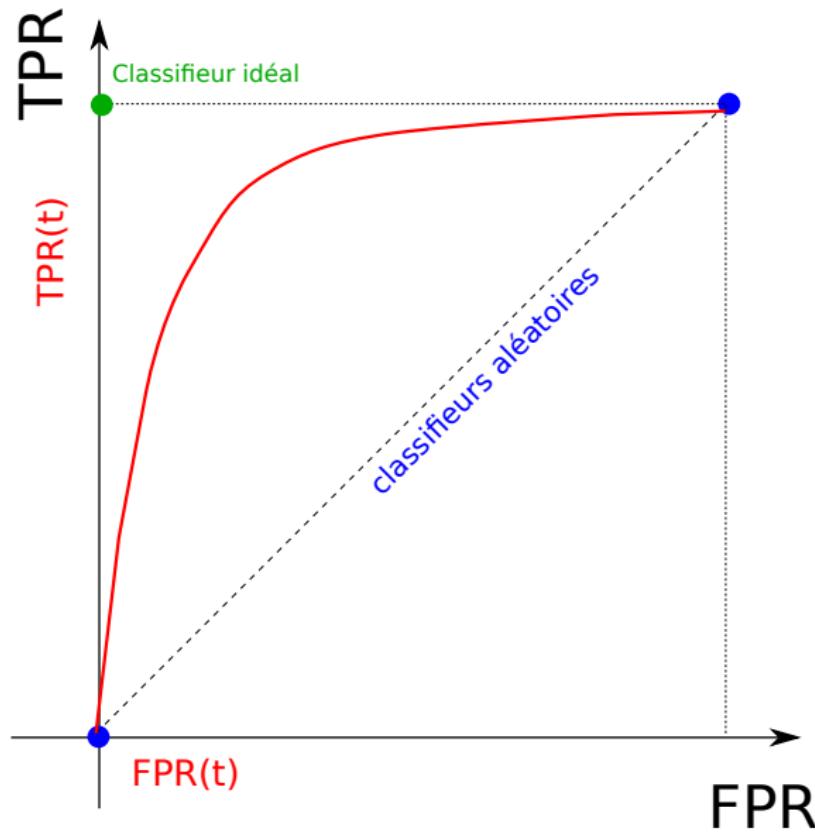
Courbe ROC



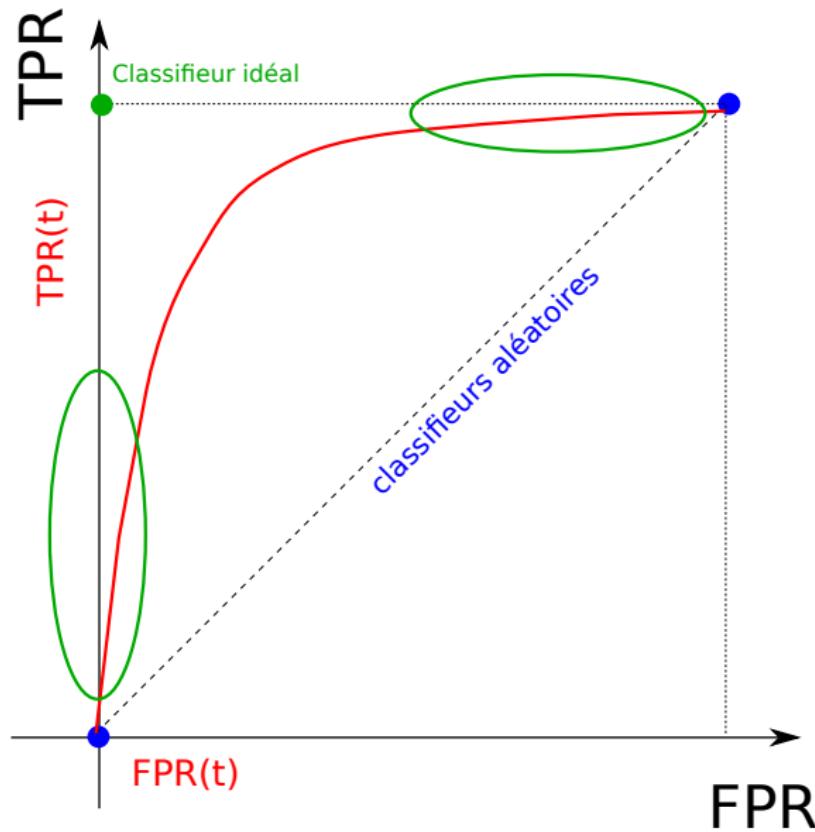
Courbe ROC



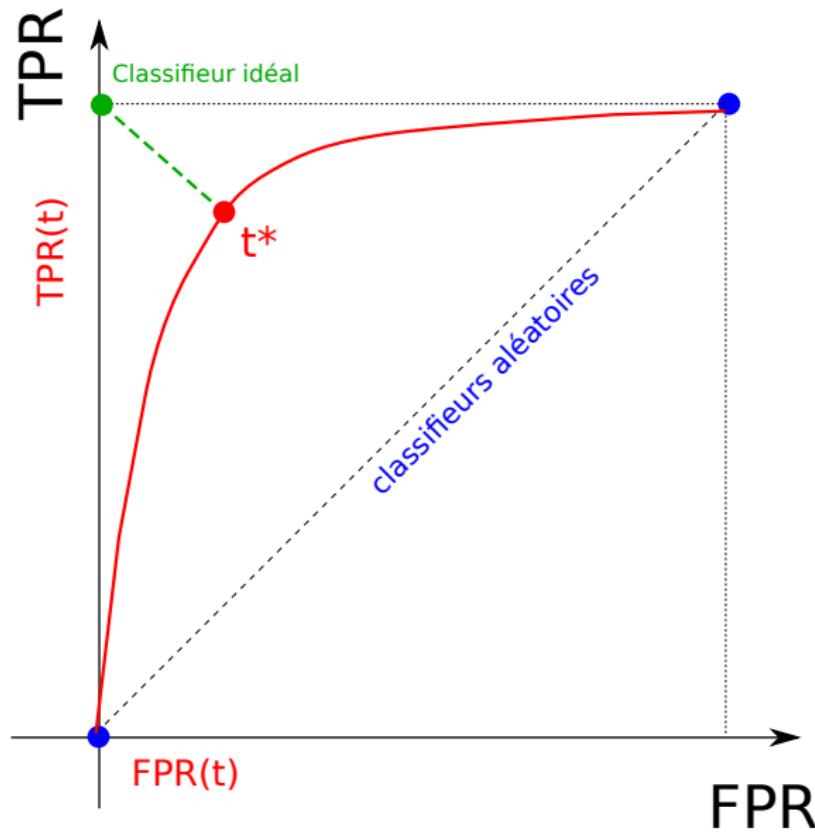
Courbe ROC



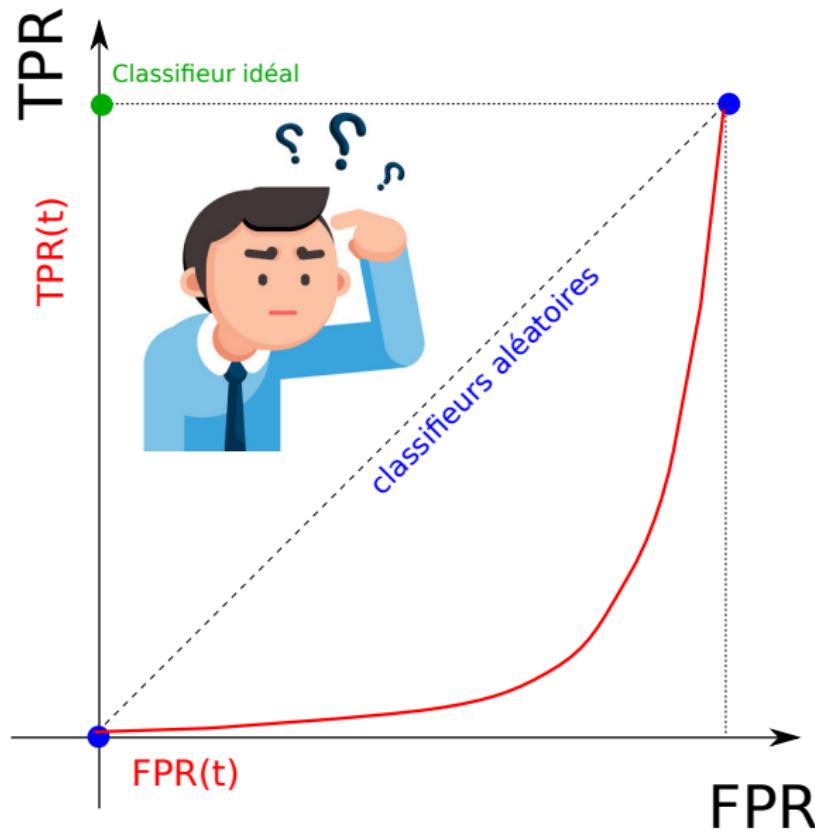
Courbe ROC



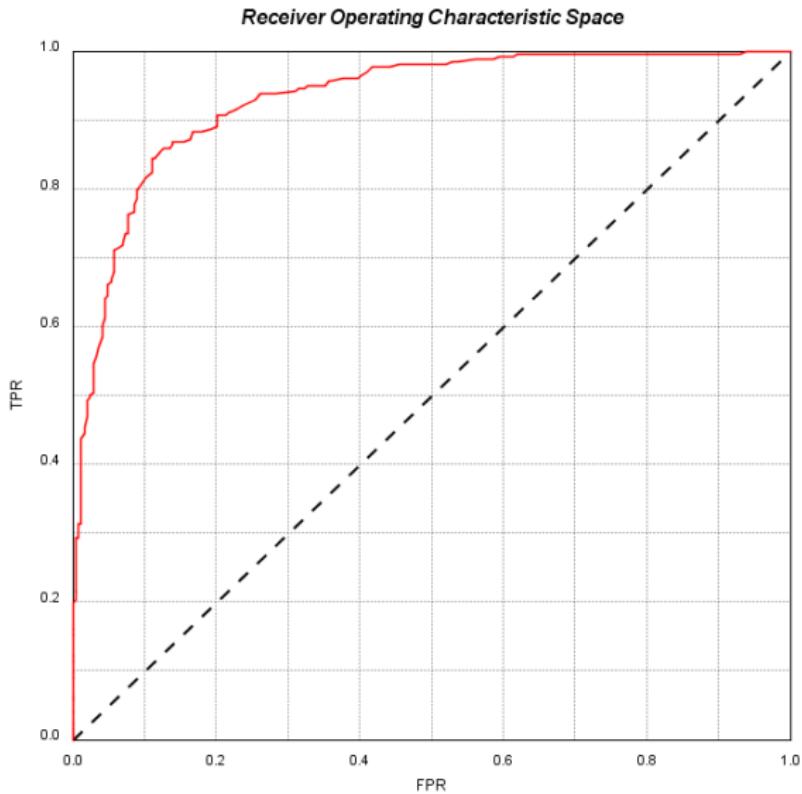
Courbe ROC



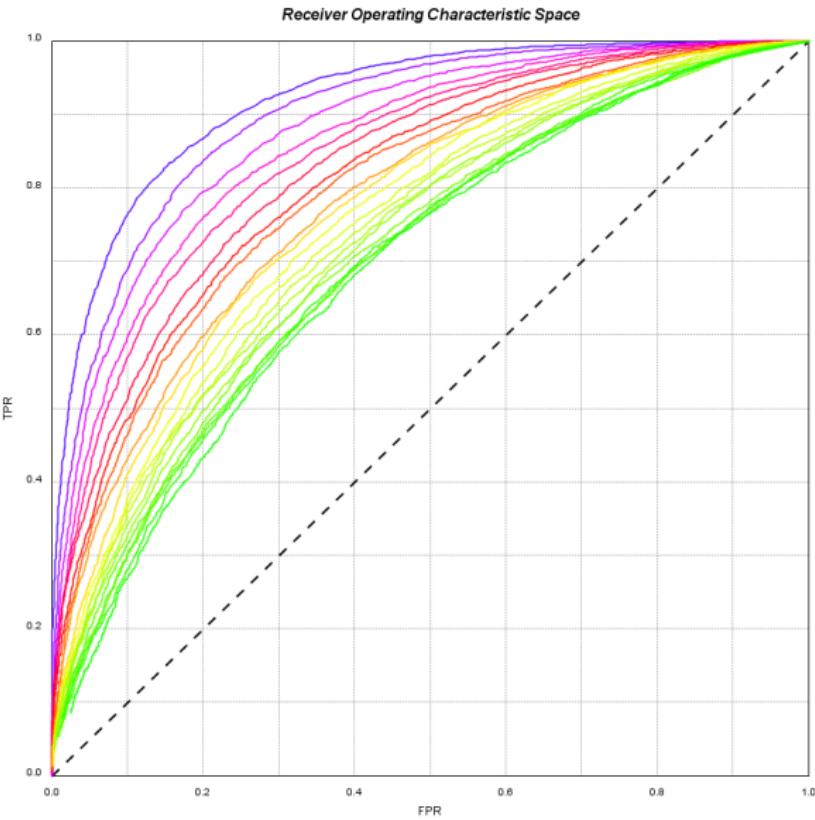
Courbe ROC



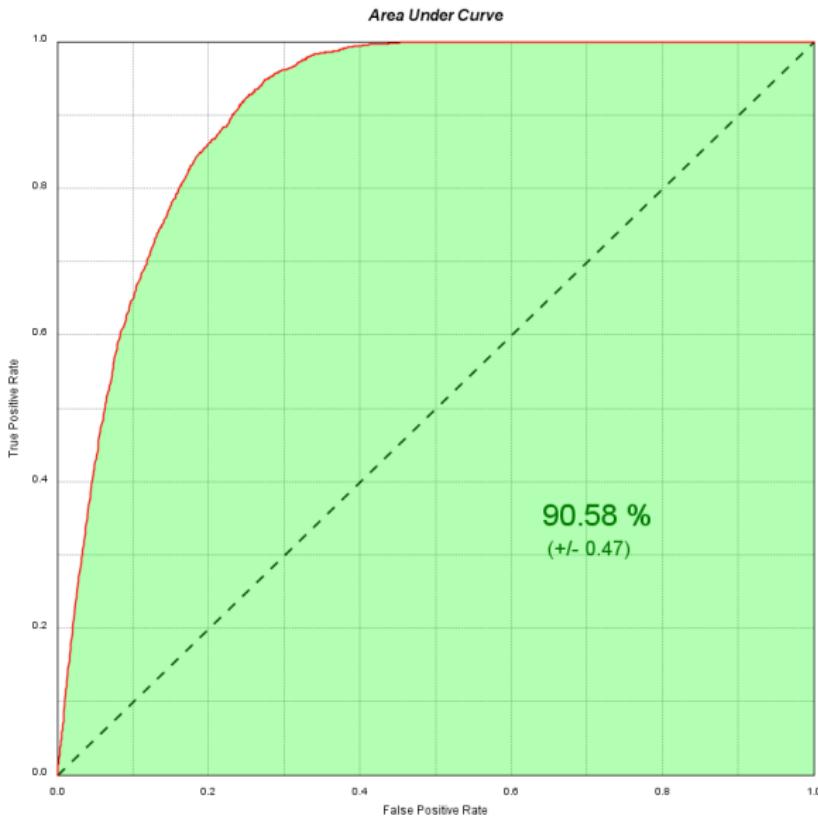
Courbe ROC



Courbe ROC



Courbe ROC



Courbe ROC

On appelle **courbe ROC** (pour *Receiver Operating Characteristics*) la courbe f paramétrée par le niveau de seuillage t , constituée de l'ensemble des points : $(F = \text{FPR}(t), T = \text{TPR}(t))$

Courbe ROC

On appelle **courbe ROC** (pour *Receiver Operating Characteristics*) la courbe f paramétrée par le niveau de seuillage t , constituée de l'ensemble des points : ($F = \text{FPR}(t)$, $T = \text{TPR}(t)$)

Aire sous la courbe

On appelle **aire sous la courbe** (AUC), l'intégrale :

$$AUC = \int_0^1 T(F)dF$$

Courbe ROC

On appelle **courbe ROC** (pour *Receiver Operating Characteristics*) la courbe f paramétrée par le niveau de seuillage t , constituée de l'ensemble des points : $(F = \text{FPR}(t), T = \text{TPR}(t))$

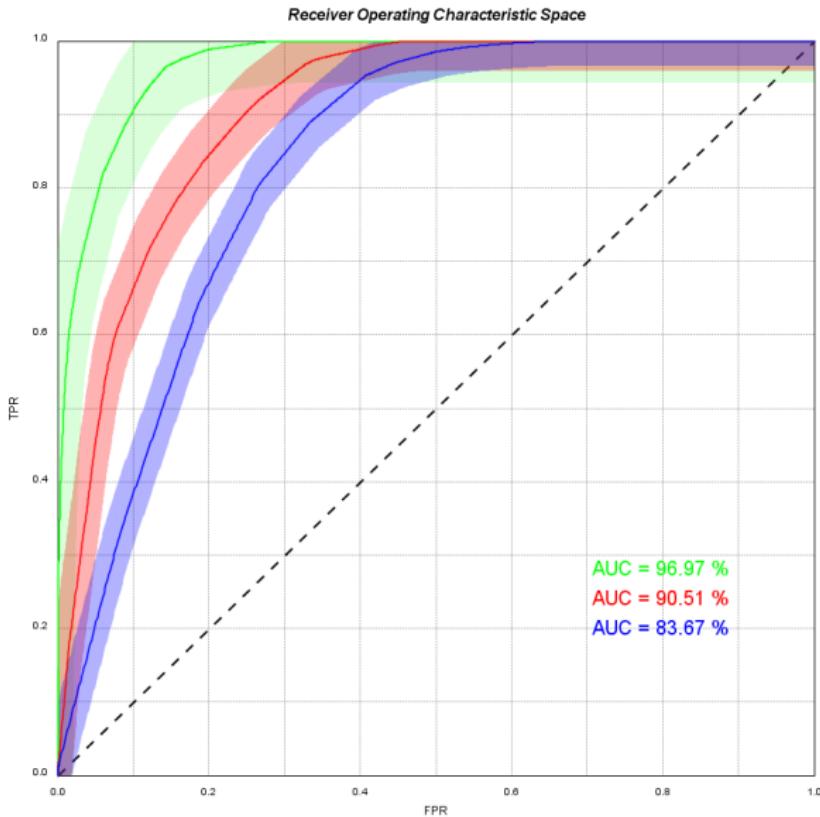
Aire sous la courbe

On appelle **aire sous la courbe** (AUC), l'intégrale :

$$AUC = \int_0^1 T(F)dF$$

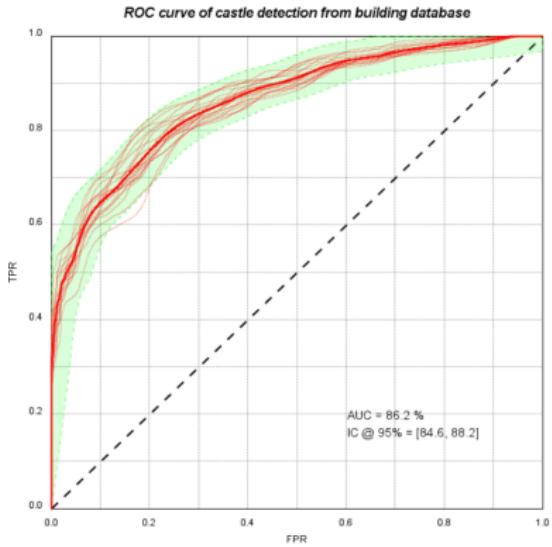
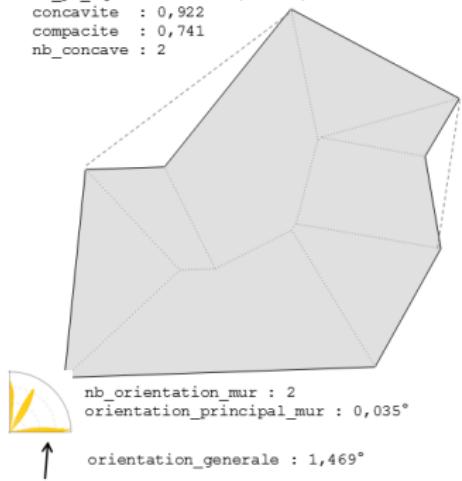
L'aire sous la courbe ROC s'interprète comme la probabilité que les affectations p_1 et p_2 sur 2 données d'étiquettes $Y_1 = 0$ et $Y_2 = 1$ soient telles que $p_1 \leq p_2$: $\mathbb{P}[p_1 \leq p_2 \mid Y_1 = 0, Y_2 = 1]$

Incertitude de la courbe ROC

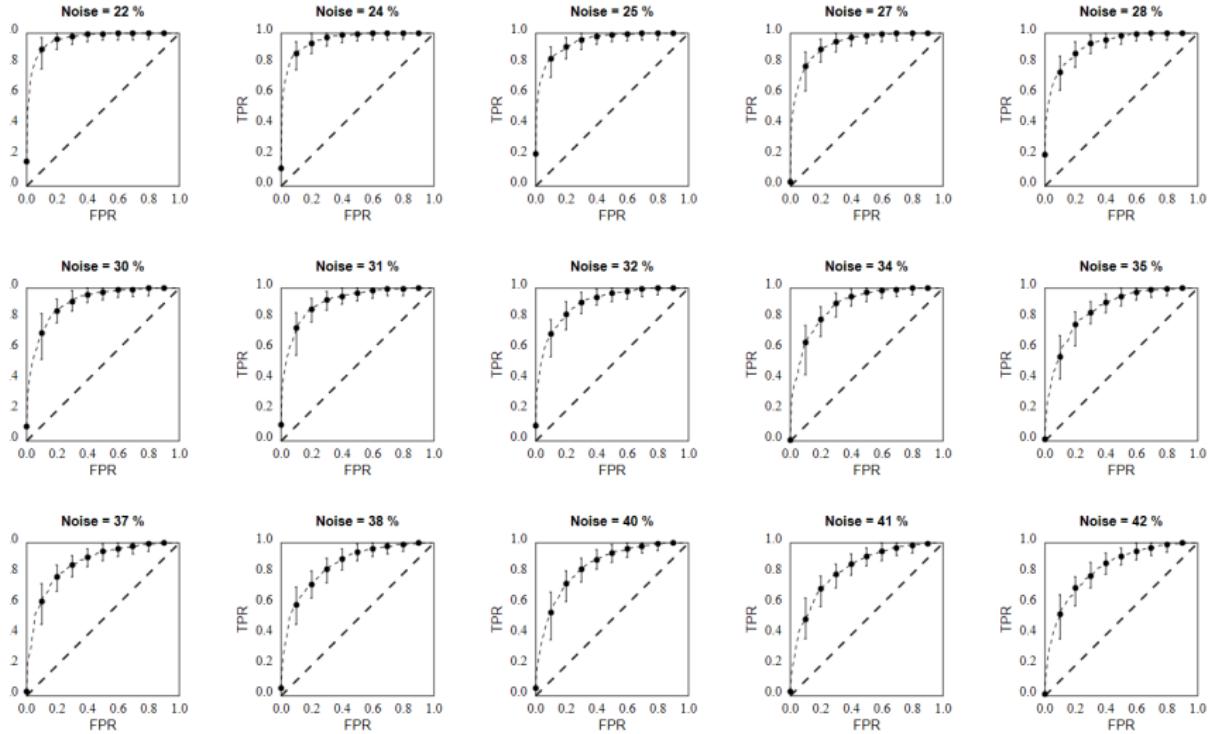


... Encore du bootstrap

```
ischateau : 1 (yes)
elongation : 0,936 (such a square)
nb_pt_squelette : 26 (vertex)
concavite : 0,922
compacite : 0,741
nb_concave : 2
```



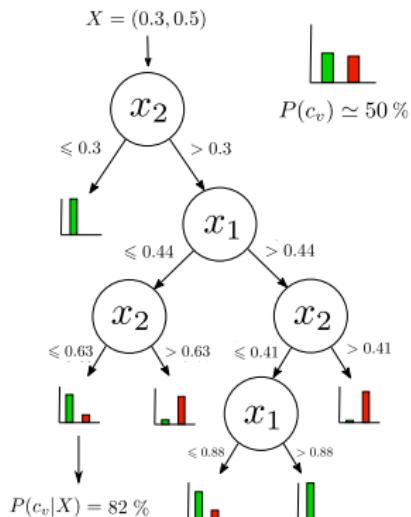
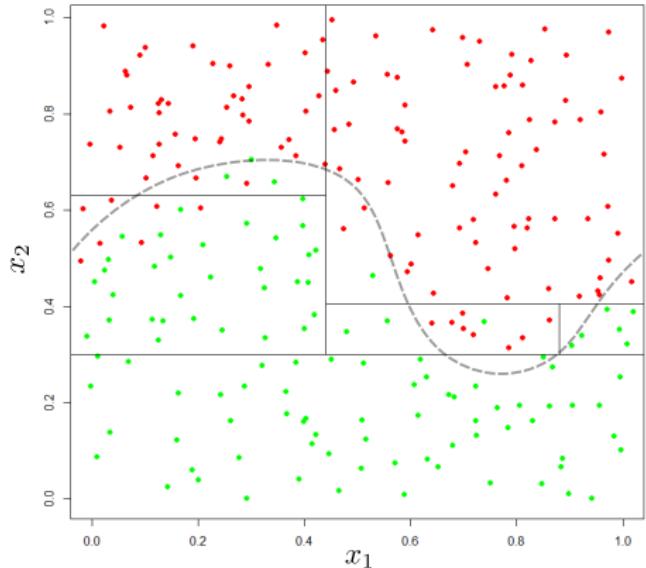
... Encore du bootstrap



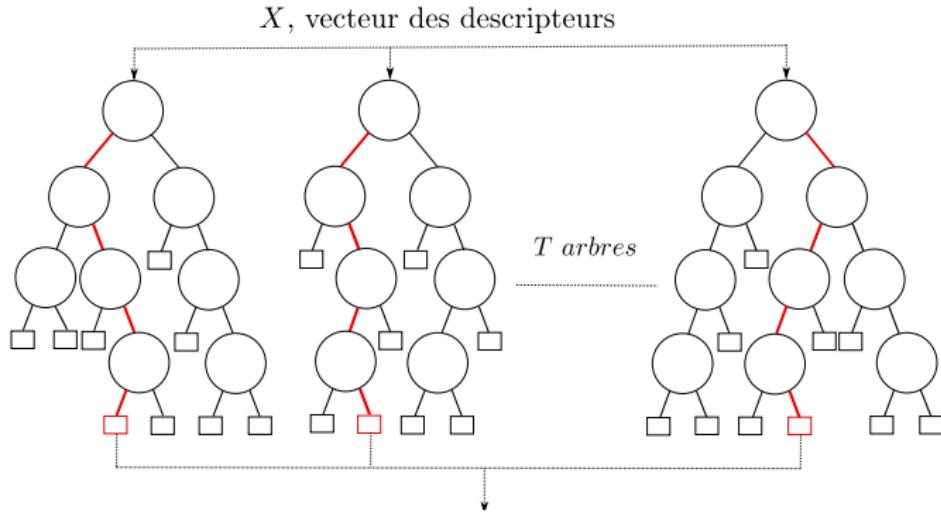
Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Arbre de décision



Forêts aléatoires (Breiman, 2001)



$$P(Y|X) = \frac{1}{N} \sum_{i=1}^N P_i(Y|X)$$

Introduction d'aleatoire à 2 niveaux :

Introduction d'aleatoire à 2 niveaux :

- À chaque coupe, on sélectionne un sous-ensemble de p descripteurs parmi tous les descripteurs disponibles ($p = \sqrt{n}$ en classification, et $p = \lceil n \rceil / 3$ en régression).

Introduction d'aleatoire à 2 niveaux :

- À chaque coupe, on sélectionne un sous-ensemble de p descripteurs parmi tous les descripteurs disponibles ($p = \sqrt{n}$ en classification, et $p = \lceil n \rceil / 3$ en régression).
- Pour chaque arbre, on sélectionne les données par tirage aléatoire avec remise dans la base d'entraînement (bootstrap).

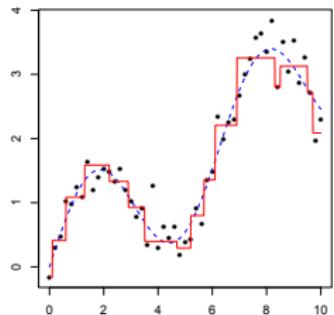
Introduction d'aleatoire à 2 niveaux :

- À chaque coupe, on sélectionne un sous-ensemble de p descripteurs parmi tous les descripteurs disponibles ($p = \sqrt{n}$ en classification, et $p = \lceil n \rceil / 3$ en régression).
- Pour chaque arbre, on sélectionne les données par tirage aléatoire avec remise dans la base d'entraînement (bootstrap).

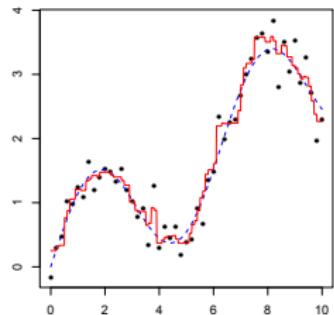
Bootstrap + Aggregation = Bagging

Forêts aléatoires (Breiman, 2001)

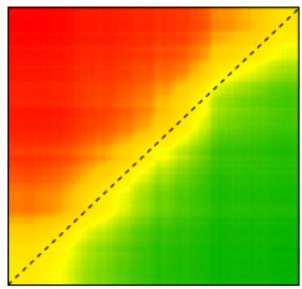
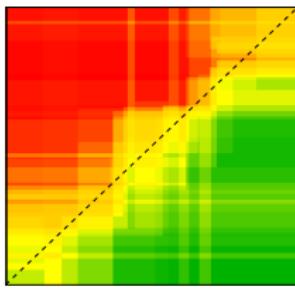
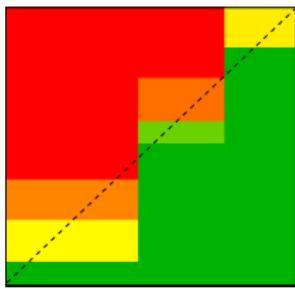
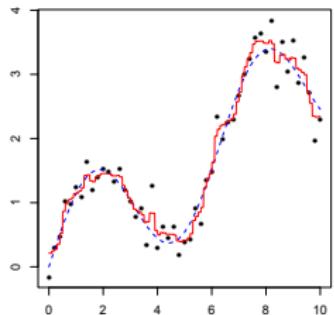
1 arbre



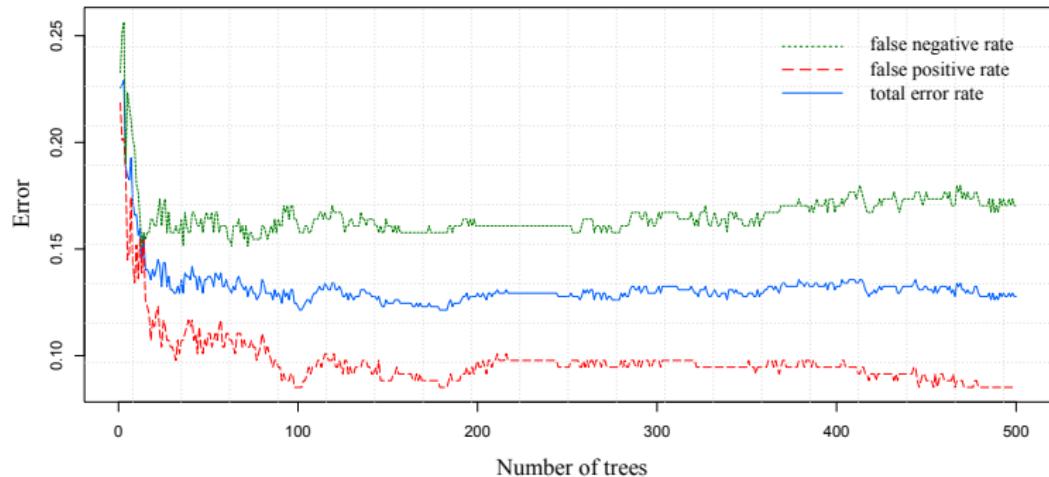
10 arbres



1000 arbres



Forêts aléatoires (Breiman, 2001)



En pratique on choisit 100 à 500 arbres.

Échantillon Out-Of-Bag (OOB) :

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée.

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée. La fraction de données non-utilisées pour la construction d'un arbre est donc :

$$\mathbb{P}[OOB] = \left(1 - \frac{1}{n}\right)^n$$

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée. La fraction de données non-utilisées pour la construction d'un arbre est donc :

$$\mathbb{P}[OOB] = \left(1 - \frac{1}{n}\right)^n$$

$$\lim_{n \sim \infty} \left(1 - \frac{1}{n}\right)^n =$$

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée. La fraction de données non-utilisées pour la construction d'un arbre est donc :

$$\mathbb{P}[OOB] = \left(1 - \frac{1}{n}\right)^n$$

$$\lim_{n \sim \infty} \left(1 - \frac{1}{n}\right)^n = e^{-1}$$

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée. La fraction de données non-utilisées pour la construction d'un arbre est donc :

$$\mathbb{P}[OOB] = \left(1 - \frac{1}{n}\right)^n$$

$$\lim_{n \sim \infty} \left(1 - \frac{1}{n}\right)^n = e^{-1} \approx 0.37$$

Échantillon Out-Of-Bag (OOB) :

Pour chaque arbre individuel, à chaque tirage bootstrap, chaque donnée a une probabilité $1 - \frac{1}{n}$ de ne pas être sélectionnée. La fraction de données non-utilisées pour la construction d'un arbre est donc :

$$\mathbb{P}[OOB] = \left(1 - \frac{1}{n}\right)^n$$

$$\lim_{n \sim \infty} \left(1 - \frac{1}{n}\right)^n = e^{-1} \approx 0.37$$

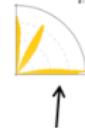
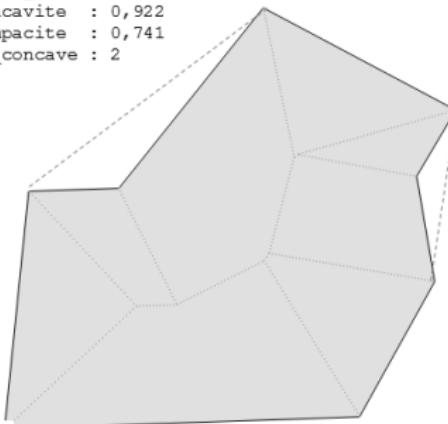
En moyenne, seules 63% des données sont utilisées pour chaque arbre. Les 37% restants peuvent être employées en **validation croisée** ! C'est l'échantillon *Out-Of-Bag*.

Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Apprentissage de données fonctionnelles

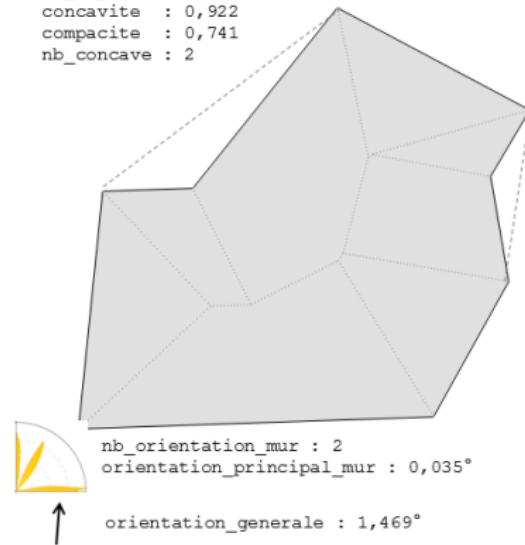
```
ischateau : 1 (yes)
elongation : 0,936 (such a square)
nb_pt_squelette : 26 (vertex)
concavite : 0,922
compacite : 0,741
nb_concave : 2
```



```
nb_orientation_mur : 2
orientation_principal_mur : 0,035°
orientation_generale : 1,469°
```

Apprentissage de données fonctionnelles

```
ischateau : 1 (yes)
elongation : 0,936 (such a square)
nb_pt_squelette : 26 (vertex)
concavite : 0,922
compacite : 0,741
nb_concave : 2
```



Quelle description pour des signaux : données vectorielles, fonctionnelles, raster, image... ?

Apprentissage de données fonctionnelles

$L^2(\Omega)$: ensemble des fonctions $f : \Omega \mapsto \mathbb{R}$ de carré intégrable.

Apprentissage de données fonctionnelles

$L^2(\Omega)$: ensemble des fonctions $f : \Omega \mapsto \mathbb{R}$ de carré intégrable.

Produit scalaire de $L^2(\Omega)$

$$\forall f, g \in L^2(\Omega) \quad \langle f, g \rangle_{L^2} = \int_{\Omega} f(x)g(x)dx$$

Apprentissage de données fonctionnelles

$L^2(\Omega)$: ensemble des fonctions $f : \Omega \mapsto \mathbb{R}$ de carré intégrable.

Produit scalaire de $L^2(\Omega)$

$$\forall f, g \in L^2(\Omega) \quad \langle f, g \rangle_{L^2} = \int_{\Omega} f(x)g(x)dx$$

$L^2(\Omega)$ est un **espace de Hilbert séparable**, il admet donc une base dénombrable $\{\varphi_i\}_{i \in \mathbb{N}}$:

Apprentissage de données fonctionnelles

$L^2(\Omega)$: ensemble des fonctions $f : \Omega \mapsto \mathbb{R}$ de carré intégrable.

Produit scalaire de $L^2(\Omega)$

$$\forall f, g \in L^2(\Omega) \quad \langle f, g \rangle_{L^2} = \int_{\Omega} f(x)g(x)dx$$

$L^2(\Omega)$ est un **espace de Hilbert séparable**, il admet donc une base dénombrable $\{\varphi_i\}_{i \in \mathbb{N}}$:

Base de Hilbert

$$\forall x \in \Omega : f(x) = \sum_{k=1}^{+\infty} \langle f, \varphi_k \rangle \varphi_k(x)$$

où $\{\varphi_k ; k \in \mathbb{N}\}$ est une base orthogonale de $L^2(\Omega)$

Apprentissage de données fonctionnelles

Pratiquement, on ne peut retenir qu'un nombre fini de descripteurs :

Apprentissage de données fonctionnelles

Pratiquement, on ne peut retenir qu'un nombre fini de descripteurs :

Troncature à l'ordre p

$$\forall x \in \Omega : f(x) = \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i(x) + \varepsilon_p(x),$$

où $X_i = \langle f, \varphi_i \rangle = \int_{\Omega} f(x) \varphi_i(x) dx \approx \sum_{k=1}^N f(\omega_k) \varphi_i(\omega_k)$ sont les descripteurs du problème d'apprentissage.

Apprentissage de données fonctionnelles

Pratiquement, on ne peut retenir qu'un nombre fini de descripteurs :

Troncature à l'ordre p

$$\forall x \in \Omega : f(x) = \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i(x) + \varepsilon_p(x),$$

où $X_i = \langle f, \varphi_i \rangle = \int_{\Omega} f(x) \varphi_i(x) dx \approx \sum_{k=1}^N f(\omega_k) \varphi_i(\omega_k)$ sont les descripteurs du problème d'apprentissage.

Théorème : optimalité de la décomposition

Soit H un espace de Hilbert, et F le sous-espace engendré par $\varphi_1, \varphi_2, \dots, \varphi_p$, deux-à-deux orthogonales. Alors, pour toute fonction $f \in H$ et pour tout système de coefficients λ_i :

$$\left\| f - \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i \right\| \leq \left\| f - \sum_{i=1}^p \lambda_i \varphi_i \right\|$$

Apprentissage de données fonctionnelles

Pratiquement, on ne peut retenir qu'un nombre fini de descripteurs :

Troncature à l'ordre p

$$\forall x \in \Omega : f(x) = \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i(x) + \varepsilon_p(x),$$

où $X_i = \langle f, \varphi_i \rangle = \int_{\Omega} f(x) \varphi_i(x) dx \approx \sum_{k=1}^N f(\omega_k) \varphi_i(\omega_k)$ sont les descripteurs du problème d'apprentissage.

Théorème : optimalité de la décomposition

Soit H un espace de Hilbert, et F le sous-espace engendré par $\varphi_1, \varphi_2, \dots, \varphi_p$, deux-à-deux orthogonales. Alors, pour toute fonction $f \in H$ et pour tout système de coefficients λ_i :

$$\left\| f - \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i \right\| \leq \left\| f - \sum_{i=1}^p \lambda_i \varphi_i \right\|$$

Apprentissage de données fonctionnelles

Pratiquement, on ne peut retenir qu'un nombre fini de descripteurs :

Troncature à l'ordre p

$$\forall x \in \Omega : f(x) = \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i(x) + \varepsilon_p(x),$$

où $X_i = \langle f, \varphi_i \rangle = \int_{\Omega} f(x) \varphi_i(x) dx \approx \sum_{k=1}^N f(\omega_k) \varphi_i(\omega_k)$ sont les descripteurs du problème d'apprentissage.

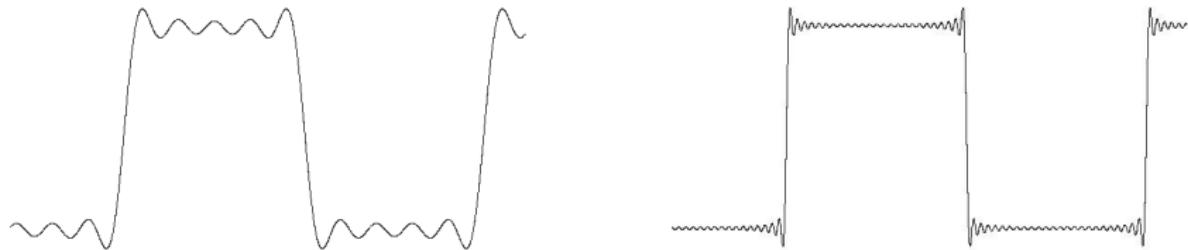
Théorème : optimalité de la décomposition

Soit H un espace de Hilbert, et F le sous-espace engendré par $\varphi_1, \varphi_2, \dots, \varphi_p$, deux-à-deux orthogonales. Alors, pour toute fonction $f \in H$ et pour tout système de coefficients λ_i :

$$\left\| f - \sum_{i=1}^p \langle f, \varphi_i \rangle \varphi_i \right\| \leq \left\| f - \sum_{i=1}^p \lambda_i \varphi_i \right\|$$

Apprentissage de données fonctionnelles

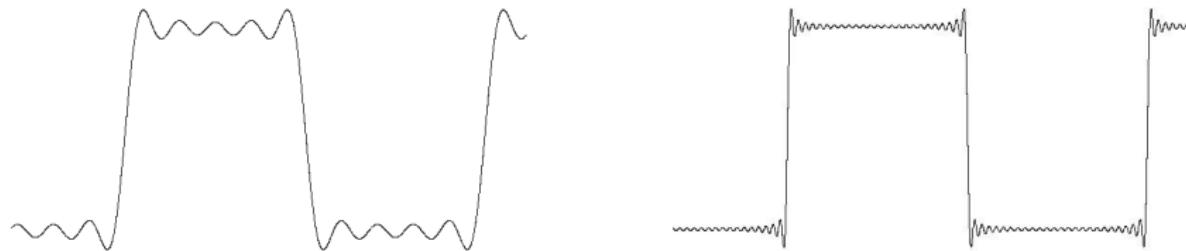
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$s(t) = K \left(\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \frac{1}{7} \sin 7\omega t + \dots \right)$$

Apprentissage de données fonctionnelles

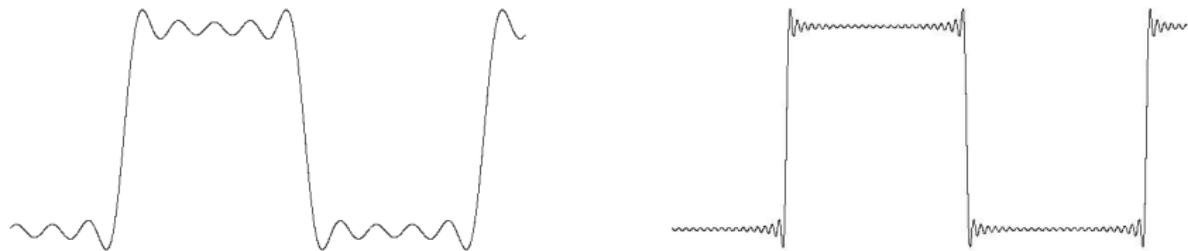
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$s(t) = K \left(\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \frac{1}{7} \sin 7\omega t + \dots \right)$$

Apprentissage de données fonctionnelles

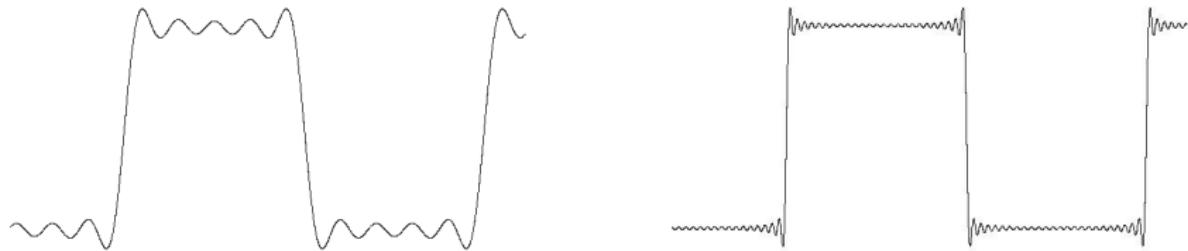
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$s(t) = K \left(1 \sin \omega t + 0 \sin 2\omega t + \frac{1}{3} \sin 3\omega t + 0 \sin 4\omega t + \frac{1}{5} \sin 5\omega t + \dots \right)$$

Apprentissage de données fonctionnelles

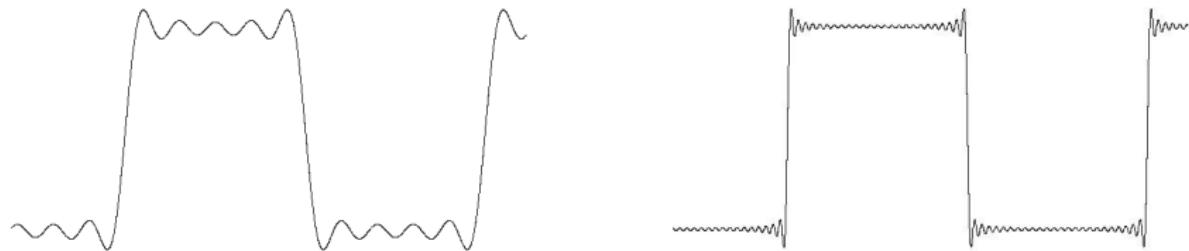
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$s(t) \approx K \left(1 \sin \omega t + 0 \sin 2\omega t + \frac{1}{3} \sin 3\omega t + 0 \sin 4\omega t + \frac{1}{5} \sin 5\omega t \right)$$

Apprentissage de données fonctionnelles

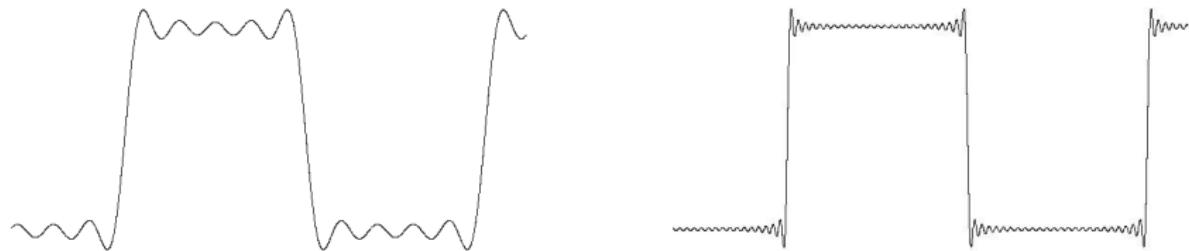
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$s \leftrightarrow K\left(1, 0, \frac{1}{3}, 0, \frac{1}{5}\right) \in \mathbb{R}^5$$

Apprentissage de données fonctionnelles

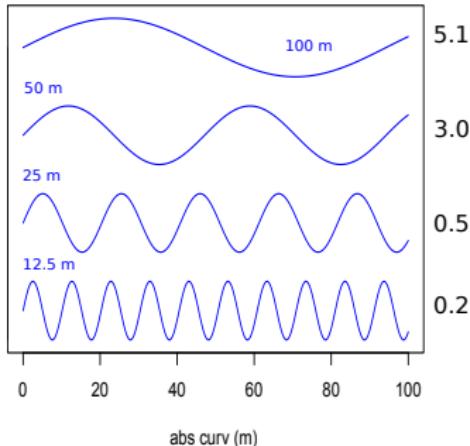
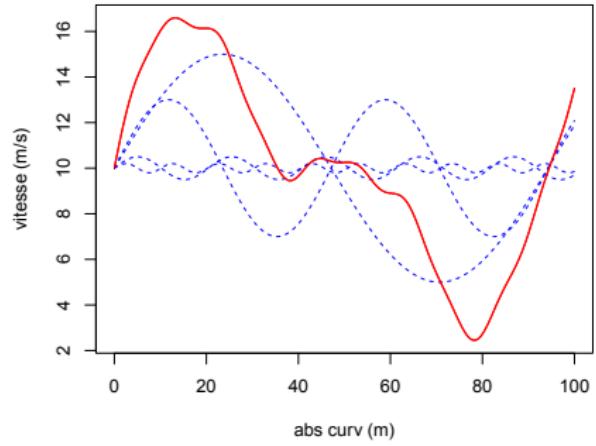
Exemple : développement en série de Fourier d'un signal créneau (prolongement périodique sur \mathbb{R})



$$L_2(\Omega) \leftrightarrow \mathbb{R}^5$$

Apprentissage de données fonctionnelles

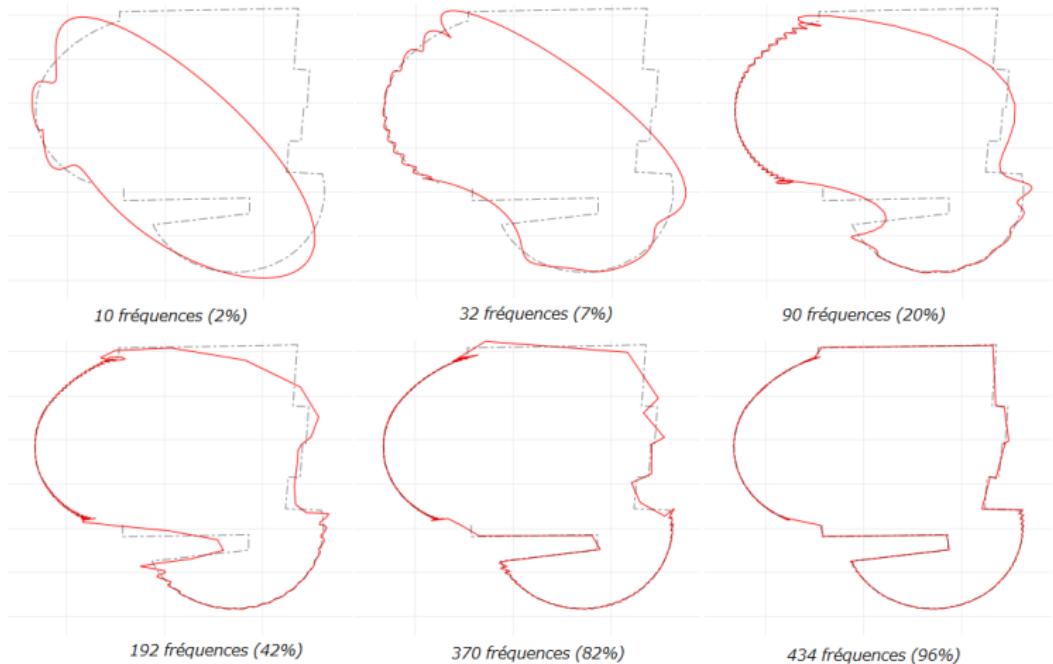
Bases fonctionnelles : Fourier, Splines, Ondelettes, KL...



<http://recherche.ign.fr/labos/cogit/demo/fourier/fourier.html>

Apprentissage de données fonctionnelles

Bases fonctionnelles : Fourier, Splines, Ondelettes, KL...



<http://recherche.ign.fr/labos/cogit/demo/fourier/fourier.html>

Apprentissage de données fonctionnelles

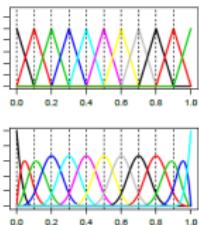
Bases fonctionnelles : Fourier, Splines, Ondelettes, KL...

$$\forall x \in [0, L] : f(x) = \sum_{i=1}^m \langle f, \varphi_i \rangle \varphi_i(x) + \varepsilon_m(x)$$

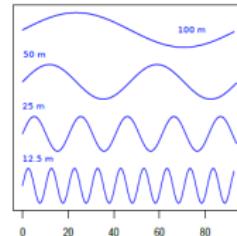
Descripteurs

Erreurs de reconstruction

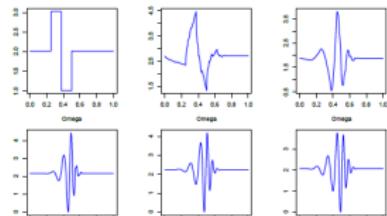
B-Splines



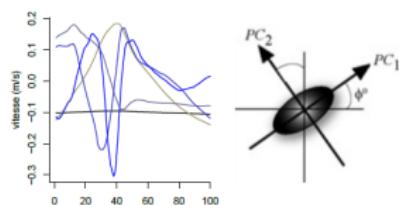
Fourier



Ondelettes



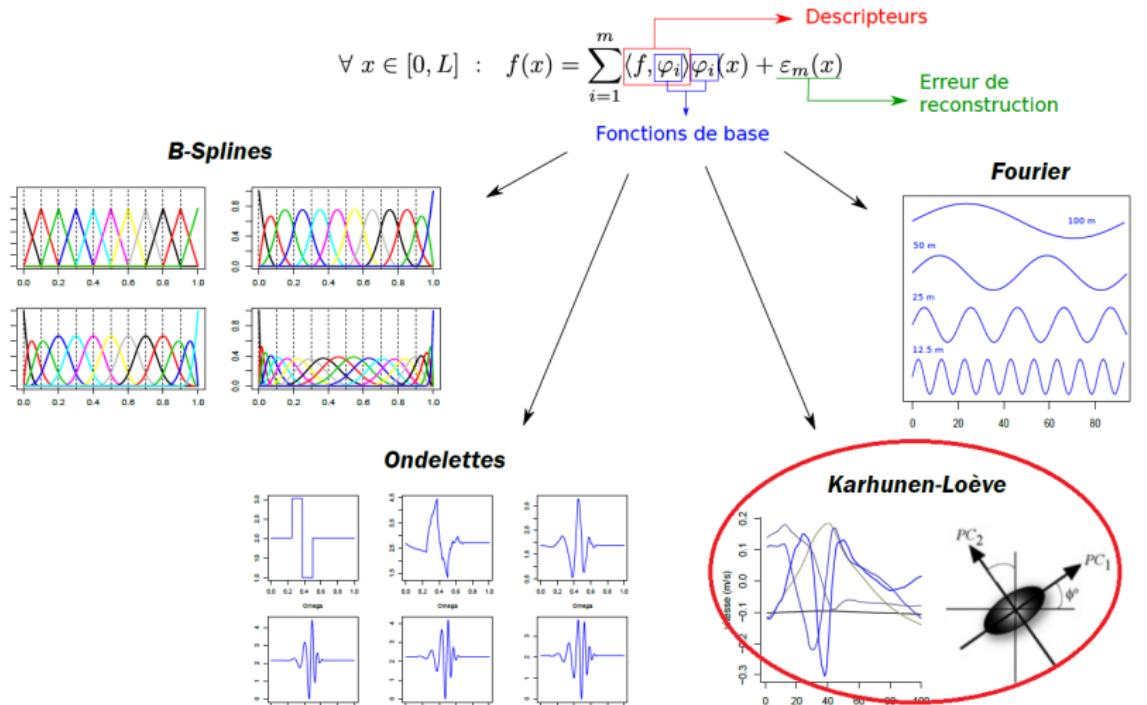
Karhunen-Loève



Projection sur une base fonctionnelle [Gregorutti, 2015]

Apprentissage de données fonctionnelles

Bases fonctionnelles : Fourier, Splines, Ondelettes, KL...



Projection sur une base fonctionnelle [Gregorutti, 2015]

Apprentissage de données fonctionnelles

Transformation de **Karhunen-Loève** (TKL) : généralisation fonctionnelle de l'ACP :

Transformation de **Karhunen-Loève** (TKL) : généralisation fonctionnelle de l'ACP : on ne travaille plus sur les réalisations $f : \mathbb{R} \mapsto \mathbb{R}$, mais sur des processus stochastiques $X : \Omega \times \mathbb{R} \mapsto \mathbb{R}$.

Théorème : optimalité de la TKL

Étant donnée une base orthonormale $\phi = \{\phi_k\}_{k \in \mathbb{N}}$ de $L^2([a, b])$, on note $\mathcal{E}_p(\phi)$ l'espérance de l'intégrale de l'erreur L_2 entre X et sa projection sur les p premiers vecteurs de la base ϕ :

$$\mathcal{E}_p(\phi) = \mathbb{E} \left[\int_a^b \left(X(\omega, t) - \sum_{i=1}^p \langle X(\omega), \phi_i \rangle \phi_i(t) \right)^2 dt \right]$$

Alors, $\mathcal{E}_p(\phi)$ est minimale si, et seulement si, les fonctions ϕ_i sont les p premières fonctions de la base issue de la TKL dans l'ordre décroissant des valeurs propres.

Apprentissage de données fonctionnelles

Rappel : si X est une variable aléatoire réelle et $a \in \mathbb{R}$ est une constante (déterministe) alors :

$$\mathbb{V}[aX] = a^2\mathbb{V}[X]$$

Apprentissage de données fonctionnelles

Rappel : si X est une variable aléatoire réelle et $a \in \mathbb{R}$ est une constante (déterministe) alors :

$$\mathbb{V}[aX] = a^2\mathbb{V}[X]$$

Similairement, en dimensions supérieures, si \mathbf{X} est un vecteur aléatoire de \mathbb{R}^n , de matrice de covariance $\Sigma_{\mathbf{X}} \in \mathbb{R}^{n \times n}$, et que \mathbf{A} est une matrice de $\mathbb{R}^{m \times n}$, quelle est la covariance de l'image \mathbf{Y} de \mathbf{X} par l'application linéaire \mathbf{A} ?

Apprentissage de données fonctionnelles

Rappel : si X est une variable aléatoire réelle et $a \in \mathbb{R}$ est une constante (déterministe) alors :

$$\mathbb{V}[aX] = a^2\mathbb{V}[X]$$

Similairement, en dimensions supérieures, si \mathbf{X} est un vecteur aléatoire de \mathbb{R}^n , de matrice de covariance $\Sigma_{\mathbf{X}} \in \mathbb{R}^{n \times n}$, et que \mathbf{A} est une matrice de $\mathbb{R}^{m \times n}$, quelle est la covariance de l'image \mathbf{Y} de \mathbf{X} par l'application linéaire \mathbf{A} ?

$$\Sigma_{\mathbf{Y}} = \mathbf{A}\Sigma_{\mathbf{X}}\mathbf{A}^T$$

Dans le cas particulier où $\mathbf{A} = a$ est un scalaire, on retrouve bien la relation : $\mathbb{V}[aX] = a^2\mathbb{V}[X]$.

Apprentissage de données fonctionnelles

(R)appel : optimisation sous contrainte.

Apprentissage de données fonctionnelles

(R)appel : optimisation sous contrainte. On considère f et g deux fonctions réelles dérivables sur $\Omega \subseteq \mathbb{R}^p$. À résoudre, le problème (\mathcal{P}) :

Apprentissage de données fonctionnelles

(R)appel : optimisation sous contrainte. On considère f et g deux fonctions réelles dérивables sur $\Omega \subseteq \mathbb{R}^p$. À résoudre, le problème (\mathcal{P}) :

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x})$$

$$\text{s.t. } g(\mathbf{x}) = 0$$

Apprentissage de données fonctionnelles

(R)appel : optimisation sous contrainte. On considère f et g deux fonctions réelles dérivables sur $\Omega \subseteq \mathbb{R}^p$. À résoudre, le problème (\mathcal{P}) :

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x})$$

$$\text{s.t. } g(\mathbf{x}) = 0$$

Théorème : multiplicateurs de Lagrange

Si un point intérieur $\mathbf{x}^* \in \Omega$ est solution de (\mathcal{P}) , alors c'est aussi un point extremum de la fonction $\mathcal{L} : \mathbb{R}^{p+1} \mapsto \mathbb{R}$ définie par :

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

Apprentissage de données fonctionnelles

(R)appel : optimisation sous contrainte. On considère f et g deux fonctions réelles dérivables sur $\Omega \subseteq \mathbb{R}^p$. À résoudre, le problème (\mathcal{P}) :

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x})$$

$$\text{s.t. } g(\mathbf{x}) = 0$$

Théorème : multiplicateurs de Lagrange

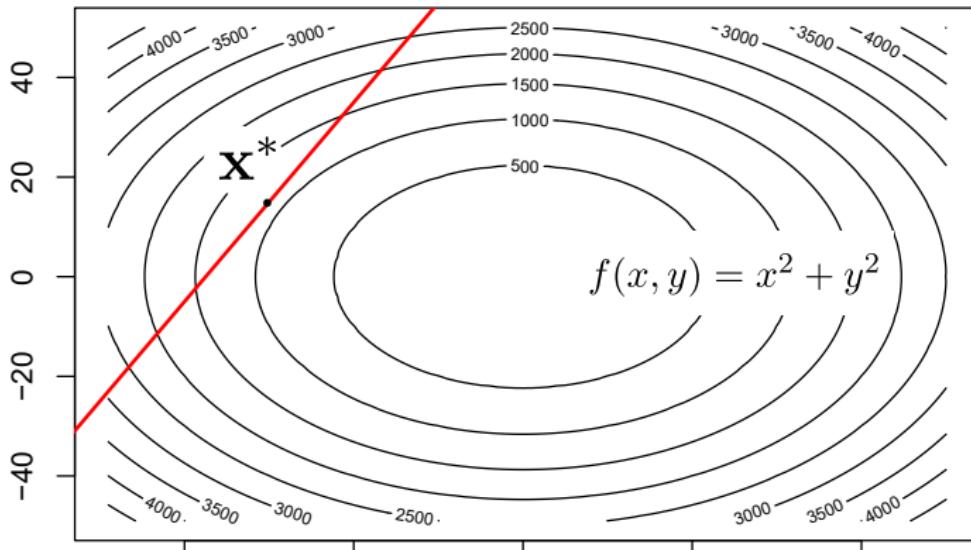
Si un point intérieur $\mathbf{x}^* \in \Omega$ est solution de (\mathcal{P}) , alors c'est aussi un point extremum de la fonction $\mathcal{L} : \mathbb{R}^{p+1} \mapsto \mathbb{R}$ définie par :

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

Remarque : $\nabla \mathcal{L}(\mathbf{x}^*) = \vec{0} \Rightarrow \nabla f(\mathbf{x}^*) = -\lambda \nabla g(\mathbf{x}^*)$ et donc le gradient de f au point solution est normal à la ligne de contrainte.

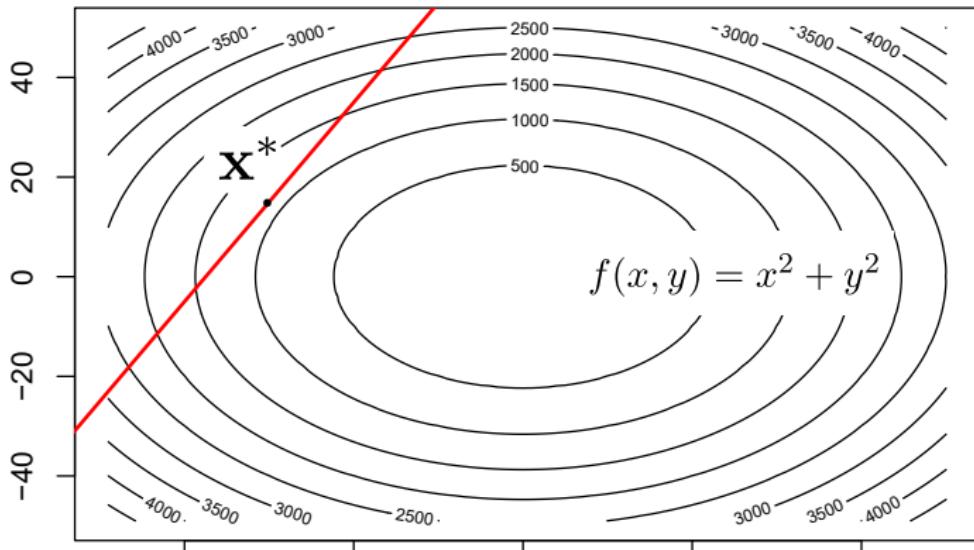
Apprentissage de données fonctionnelles

$$g(x, y) = y - 2x - 75$$



Apprentissage de données fonctionnelles

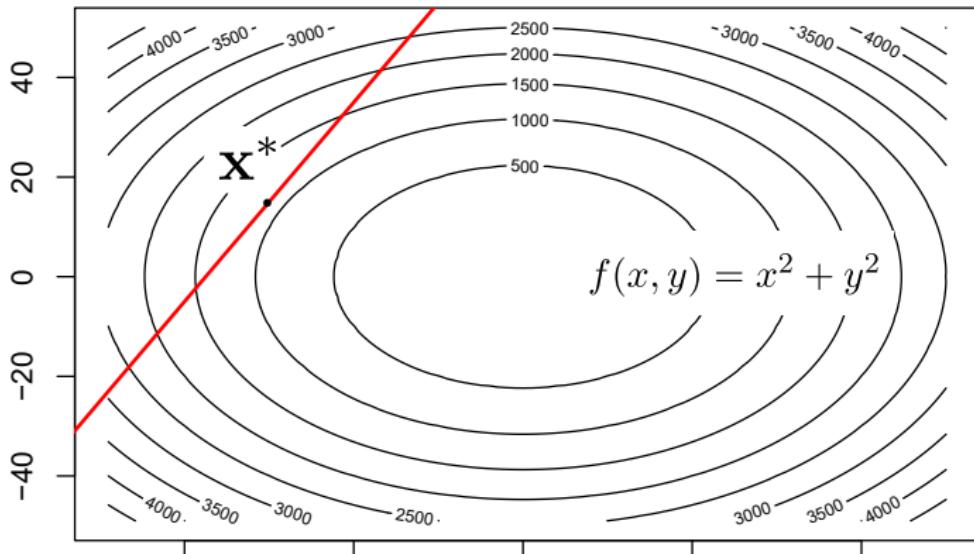
$$g(x, y) = y - 2x - 75$$



$$\mathcal{L}(x, y, \lambda) = f(x, y) + \lambda g(x, y)$$

Apprentissage de données fonctionnelles

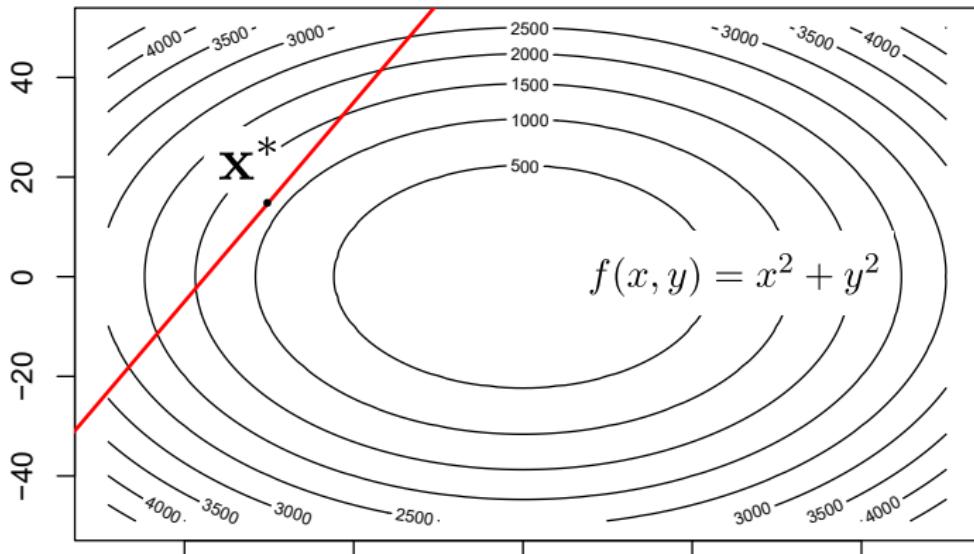
$$g(x, y) = y - 2x - 75$$



$$\mathcal{L}(x, y, \lambda) = x^2 + y^2 + \lambda(y - 2x - 75)$$

Apprentissage de données fonctionnelles

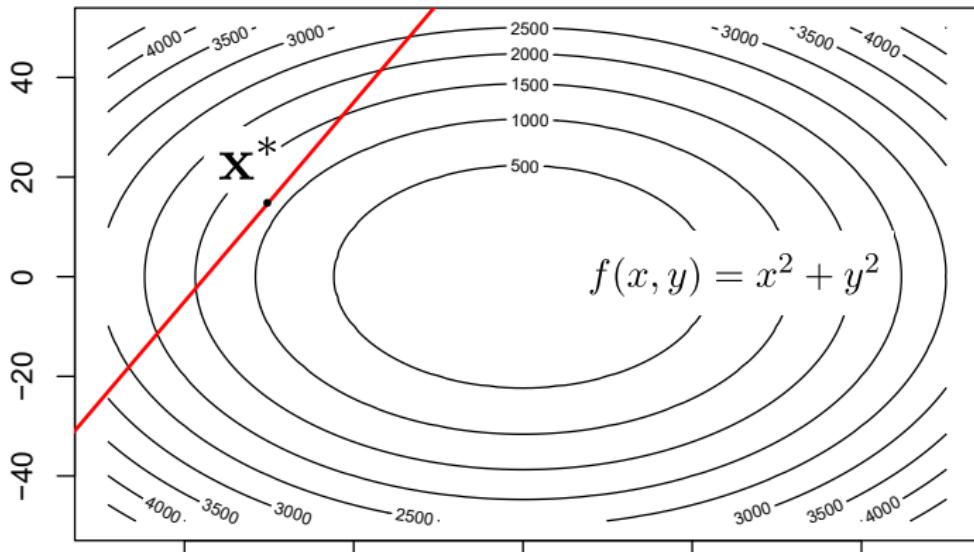
$$g(x, y) = y - 2x - 75$$



$$\nabla \mathcal{L} = \begin{cases} \partial \mathcal{L} / \partial x(\mathbf{x}^*) = 2x^* - 2\lambda^* &= 0 \\ \partial \mathcal{L} / \partial y(\mathbf{x}^*) = 2y^* + \lambda^* &= 0 \end{cases}$$

Apprentissage de données fonctionnelles

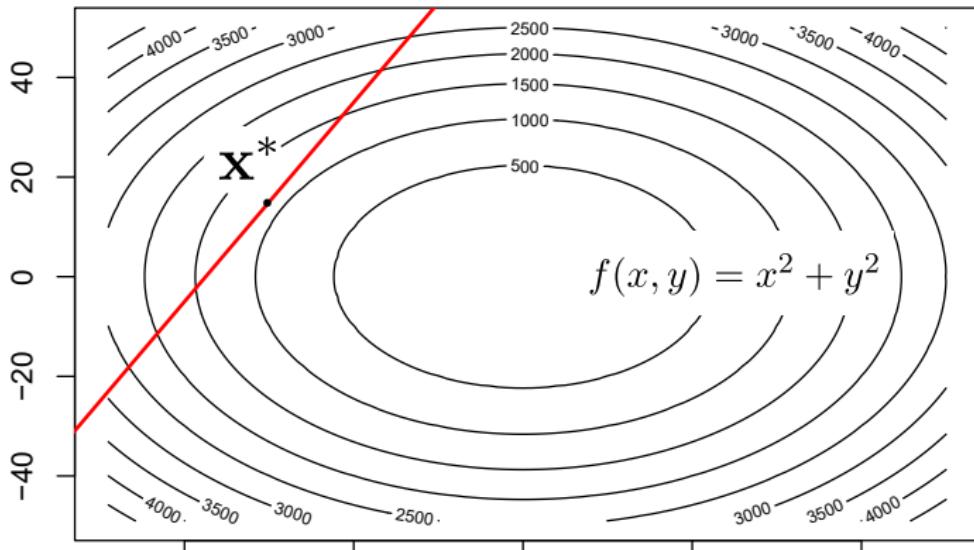
$$g(x, y) = y - 2x - 75$$



$$\begin{cases} \lambda &= x^* \\ \lambda &= -2y^* \end{cases}$$

Apprentissage de données fonctionnelles

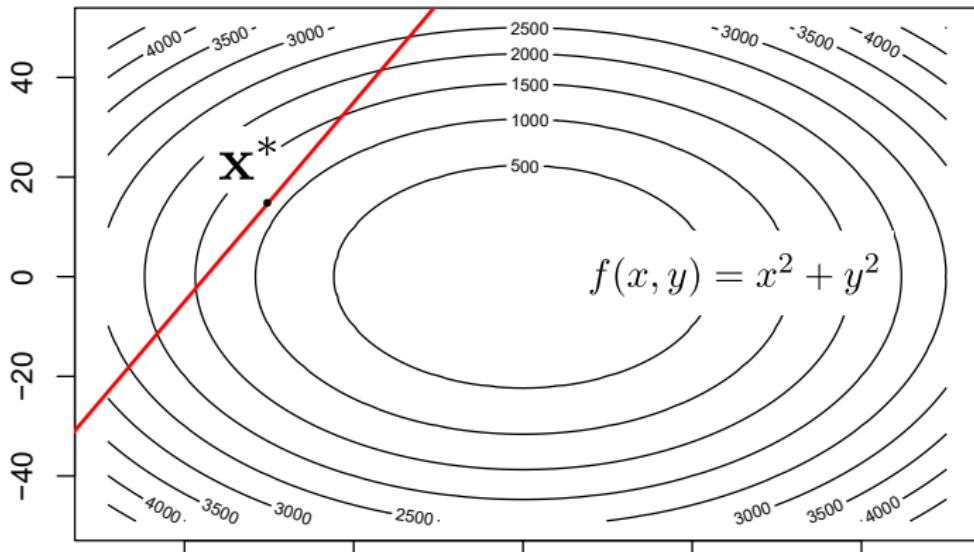
$$g(x, y) = y - 2x - 75$$



$$\begin{cases} x^* + 2y^* = 0 \\ -2x^* + y = 75 \end{cases}$$

Apprentissage de données fonctionnelles

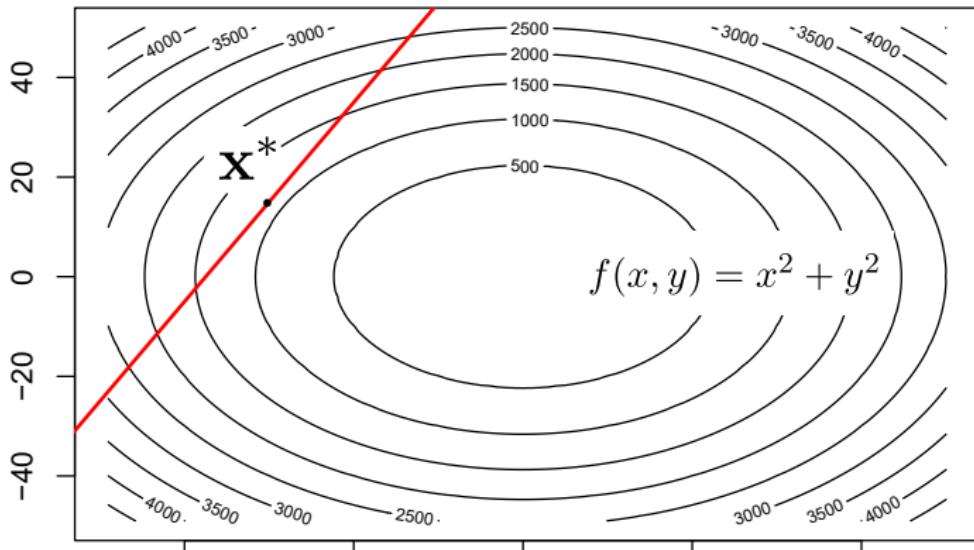
$$g(x, y) = y - 2x - 75$$



$$\mathbf{x}^* = (x^* = -30, y^* = 15, \lambda^* = -30)$$

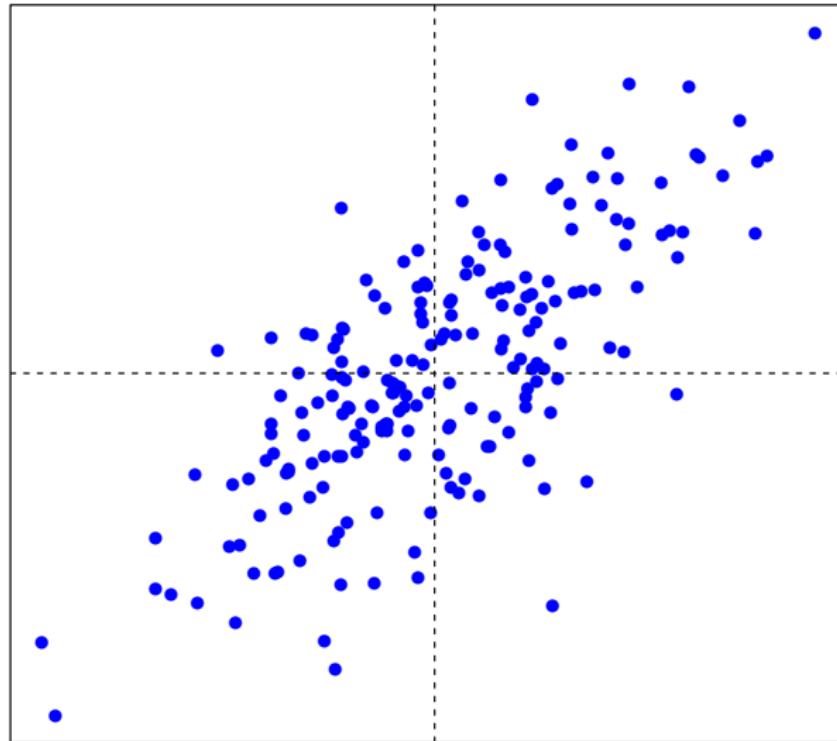
Apprentissage de données fonctionnelles

$$g(x, y) = y - 2x - 75$$

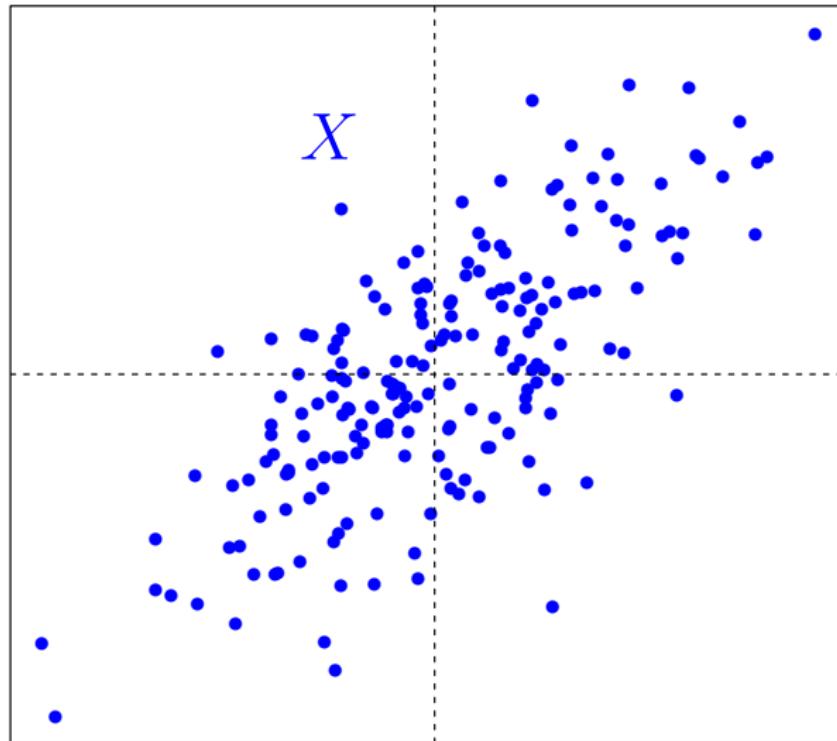


$$\mathbf{x}^* = (x^* = -30, y^* = 15, \lambda^* = -30)$$

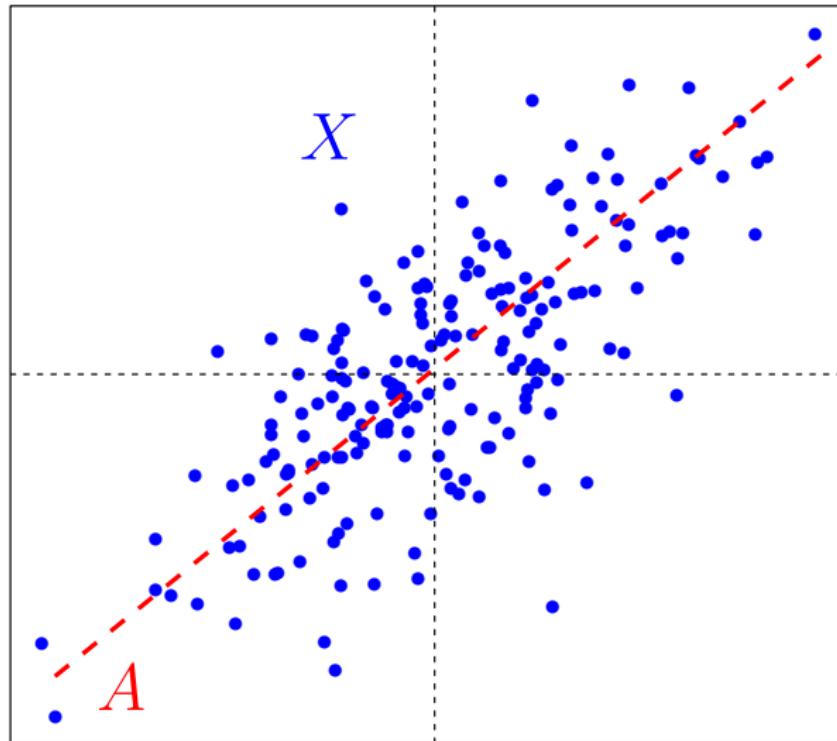
Une démonstration de la TKL en 3 lignes...



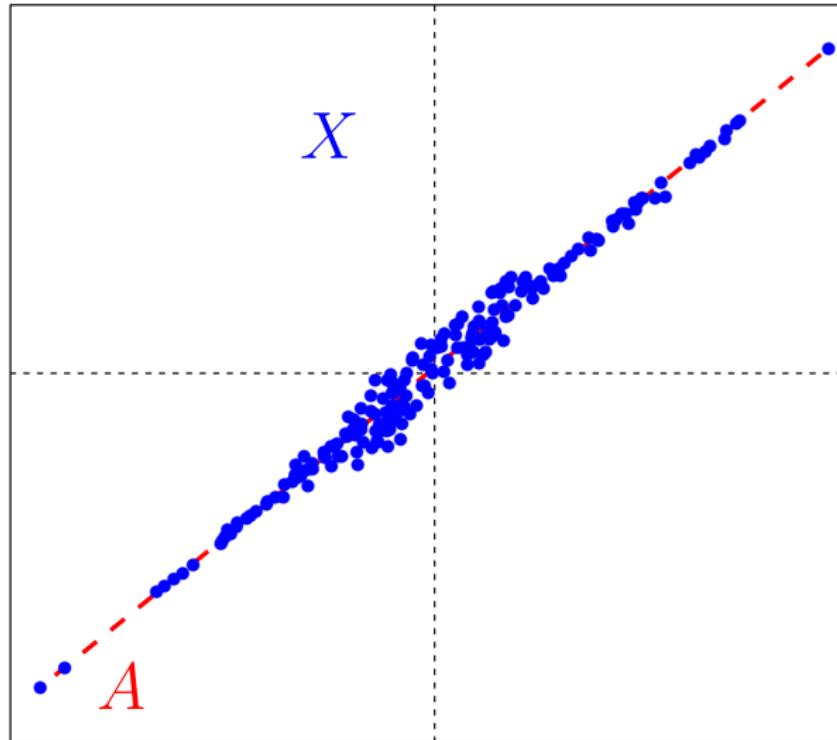
Une démonstration de la TKL en 3 lignes...



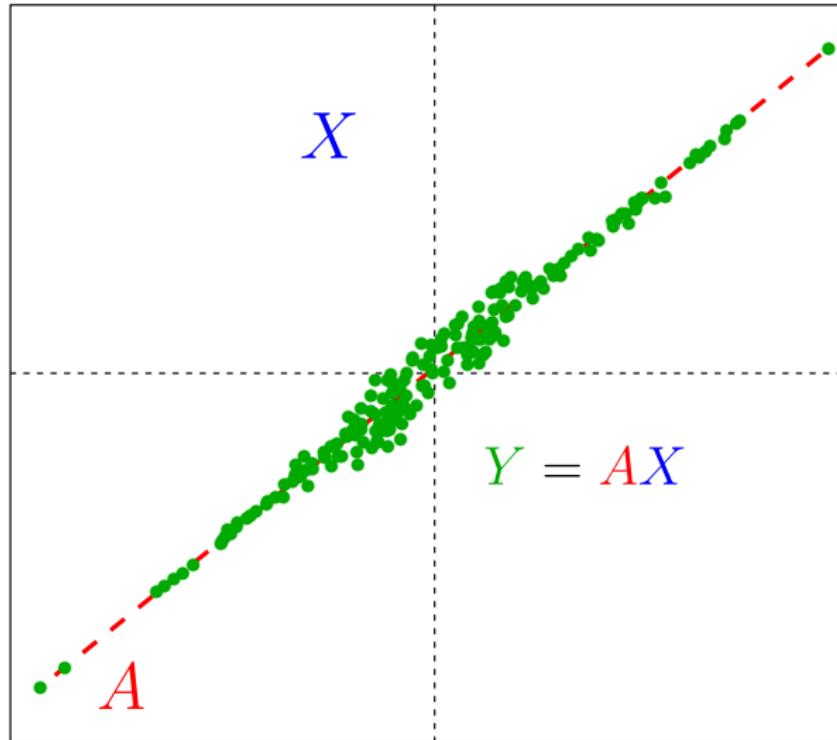
Une démonstration de la TKL en 3 lignes...



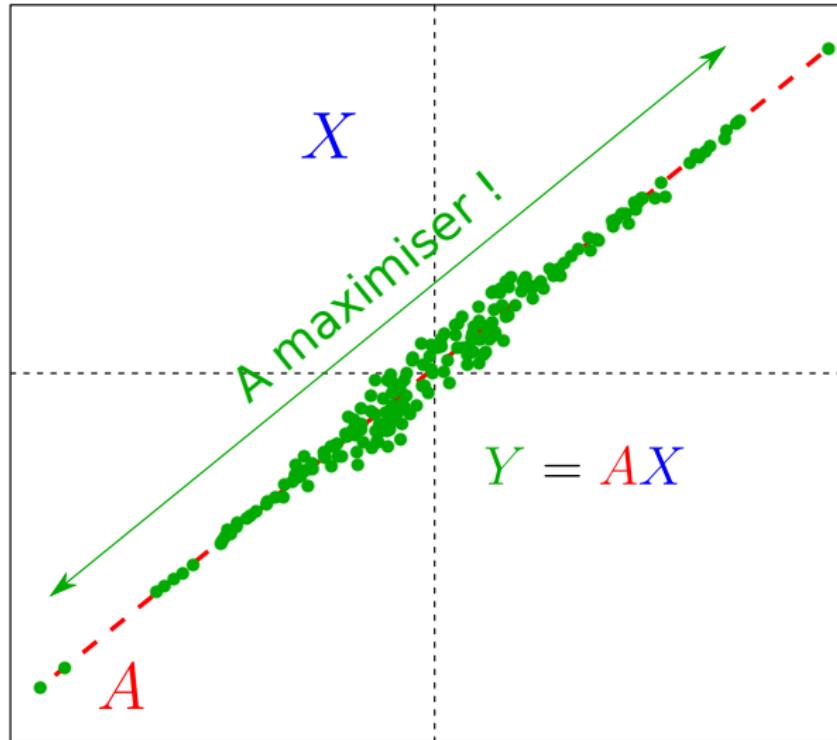
Une démonstration de la TKL en 3 lignes...



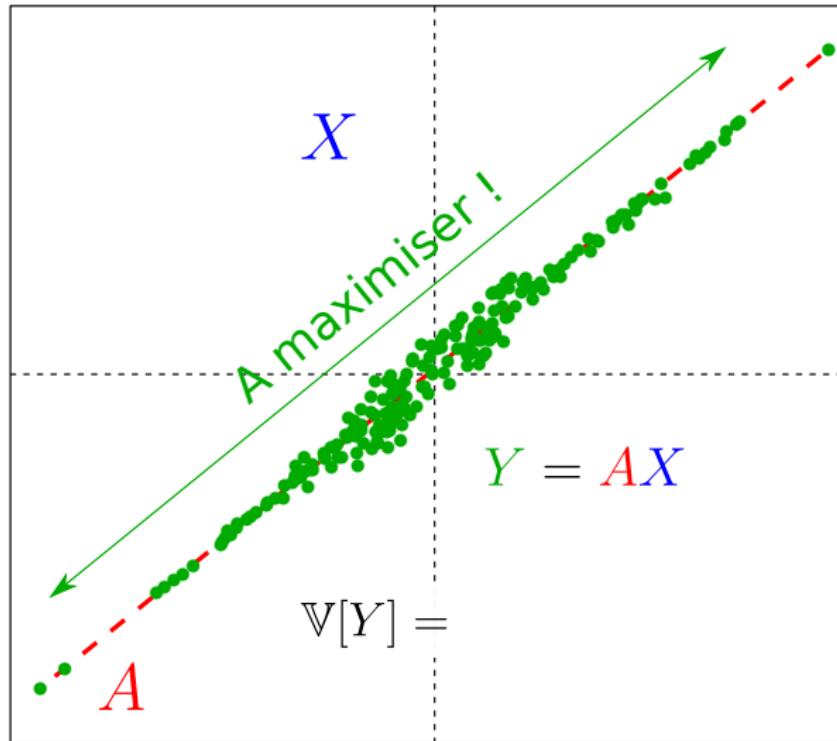
Une démonstration de la TKL en 3 lignes...



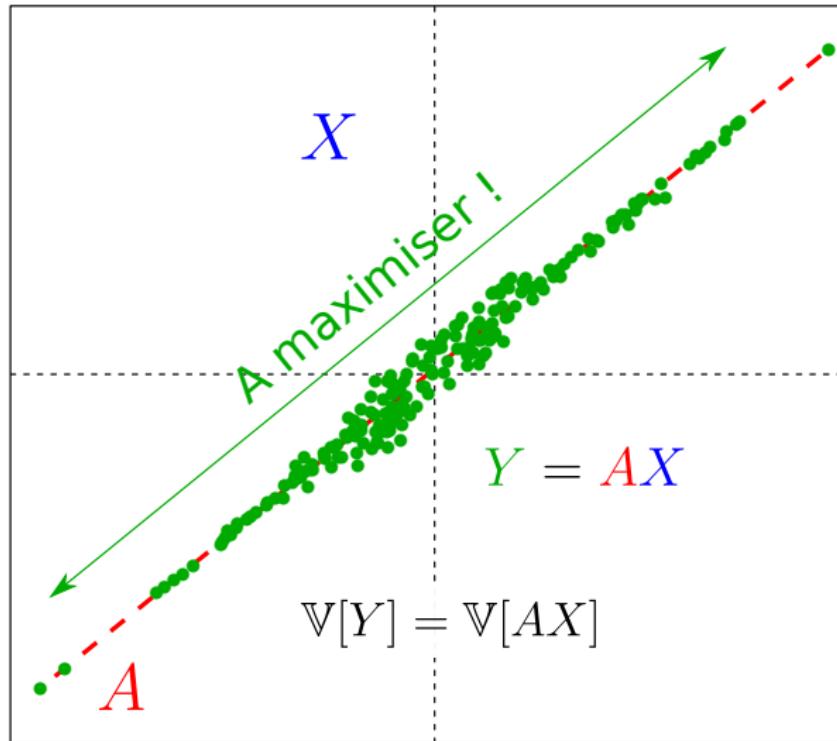
Une démonstration de la TKL en 3 lignes...



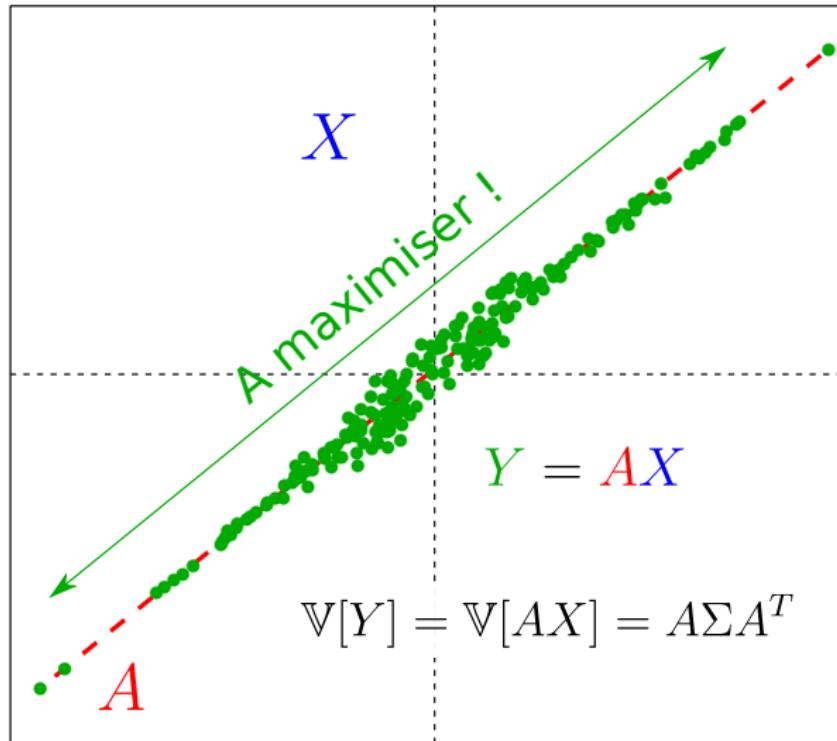
Une démonstration de la TKL en 3 lignes...



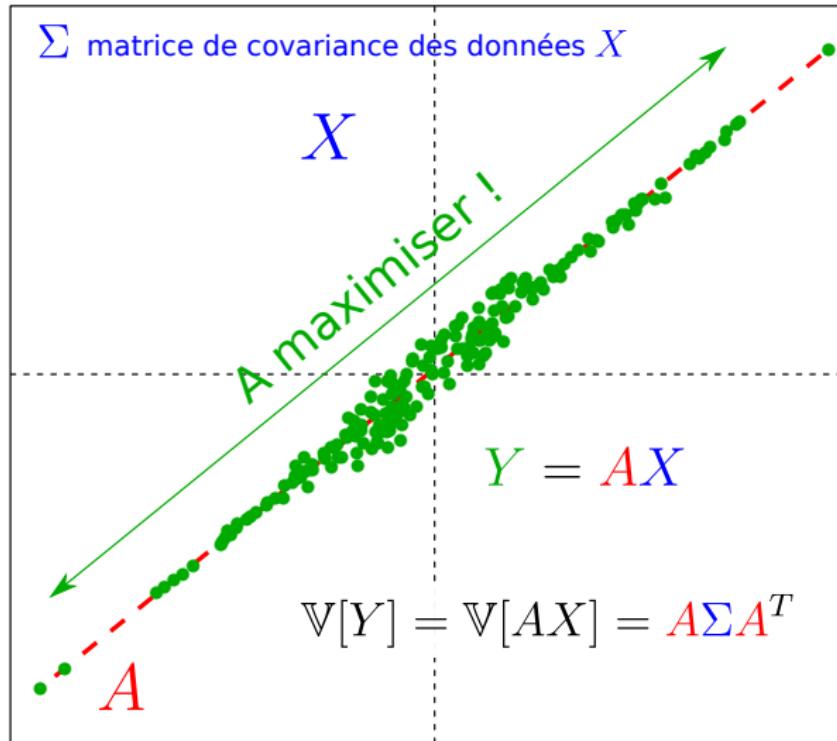
Une démonstration de la TKL en 3 lignes...



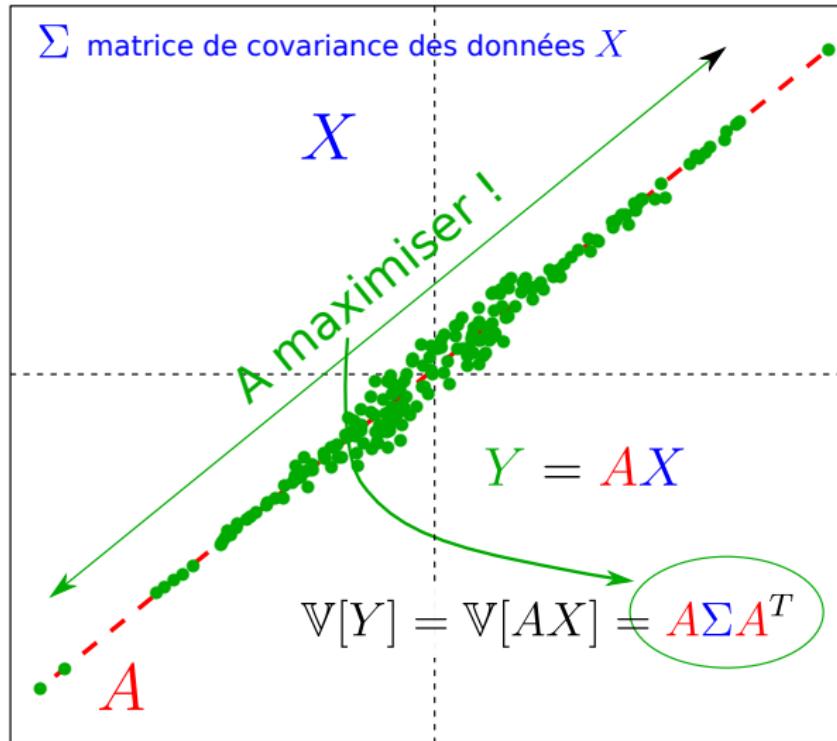
Une démonstration de la TKL en 3 lignes...



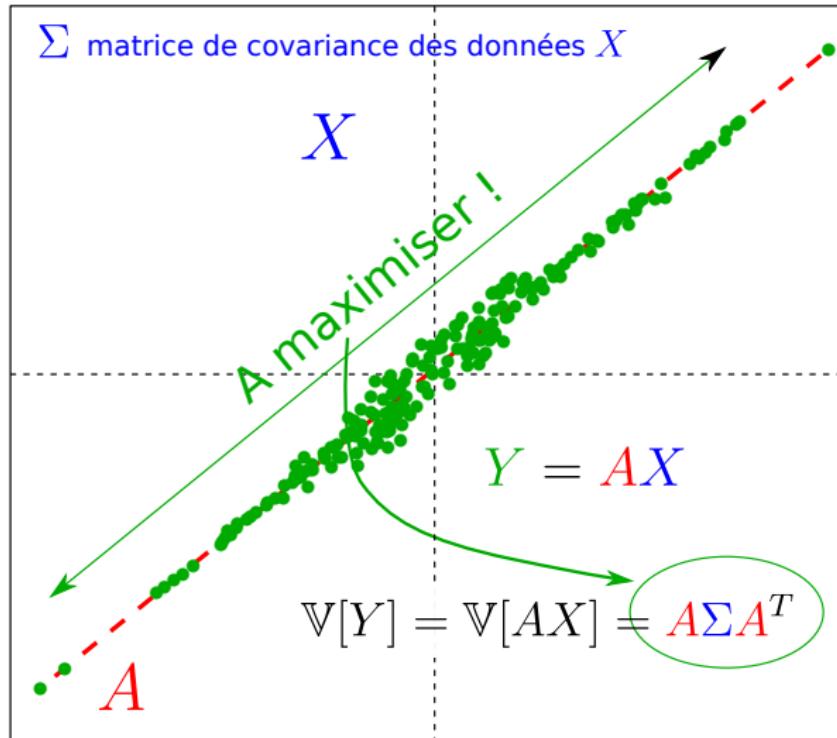
Une démonstration de la TKL en 3 lignes...



Une démonstration de la TKL en 3 lignes...

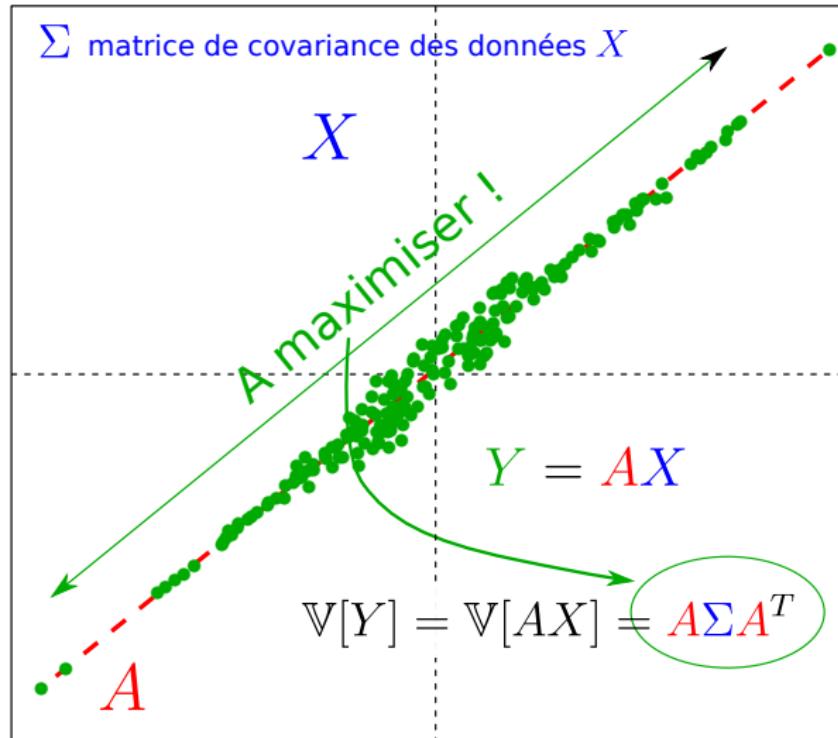


Une démonstration de la TKL en 3 lignes...



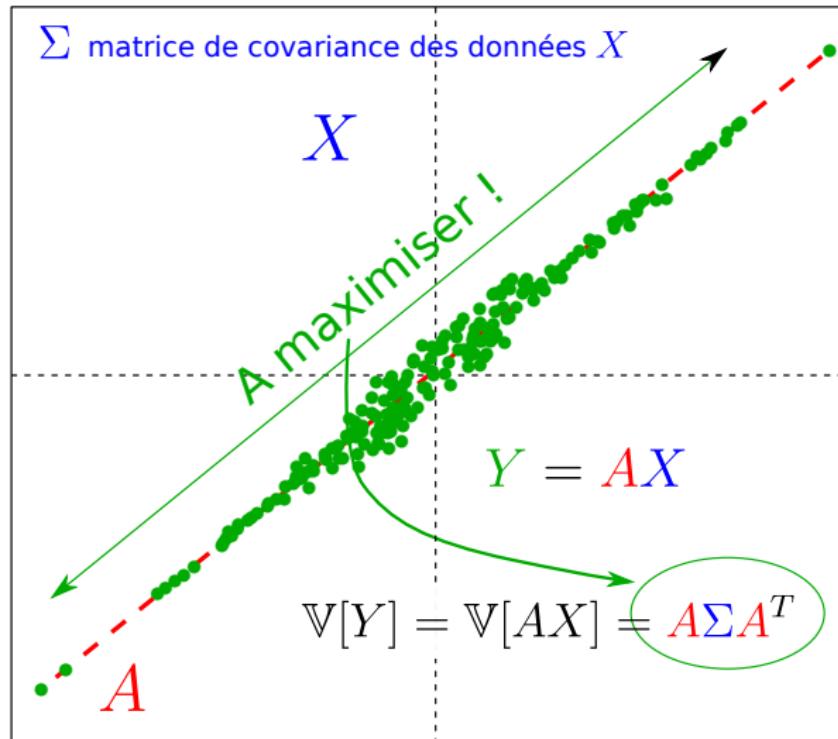
$$\max A\Sigma A^T$$

Une démonstration de la TKL en 3 lignes...



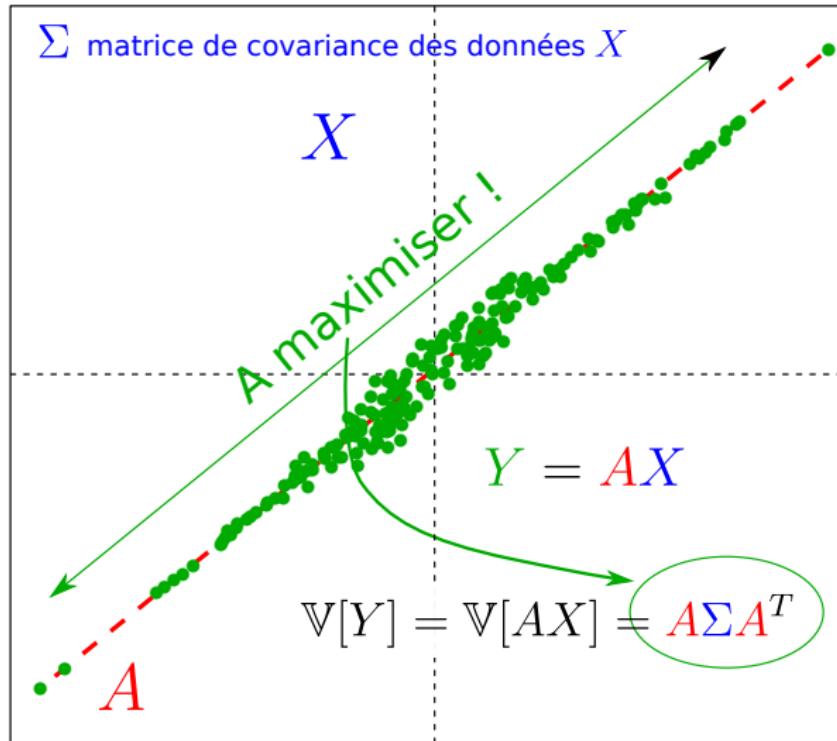
$$\max A\Sigma A^T \quad s.t. \quad \|A\| = 1$$

Une démonstration de la TKL en 3 lignes...



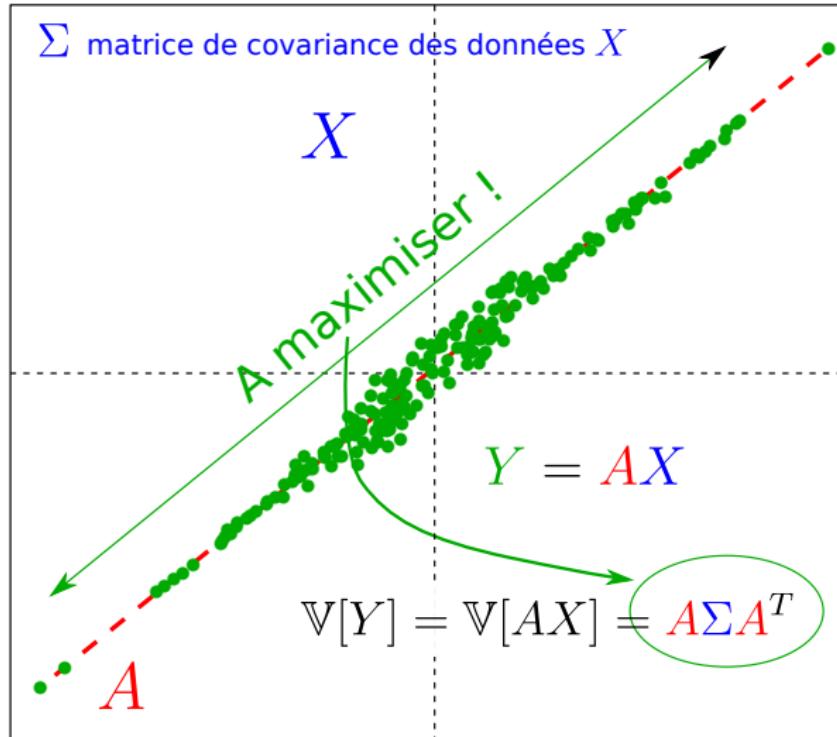
$$\max A\Sigma A^T \quad s.t. \quad AA^T = 1$$

Une démonstration de la TKL en 3 lignes...



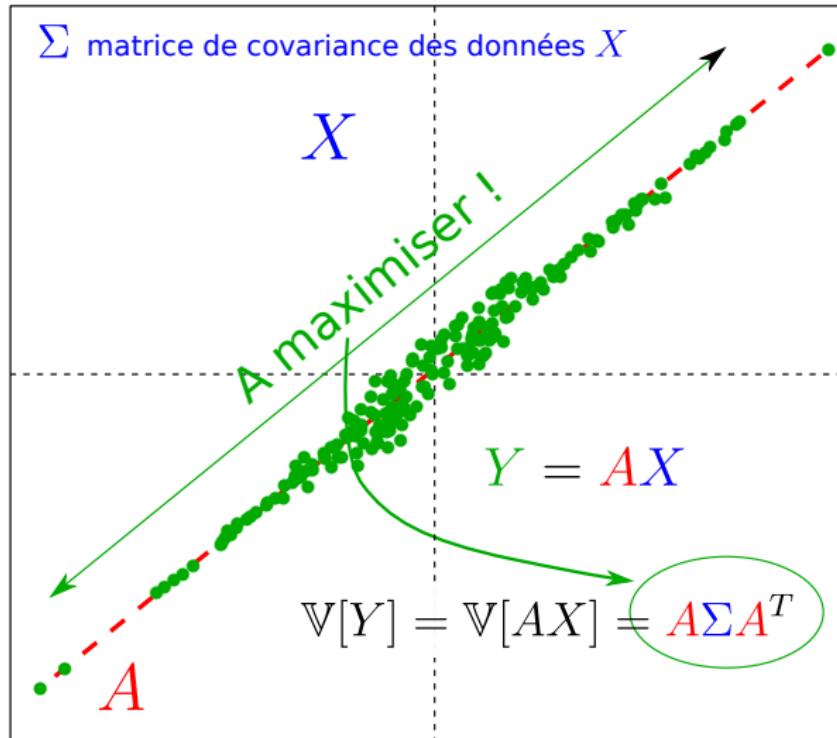
$$\mathcal{L}(A, \lambda) = A\Sigma A^T - \lambda(AA^T - 1)$$

Une démonstration de la TKL en 3 lignes...



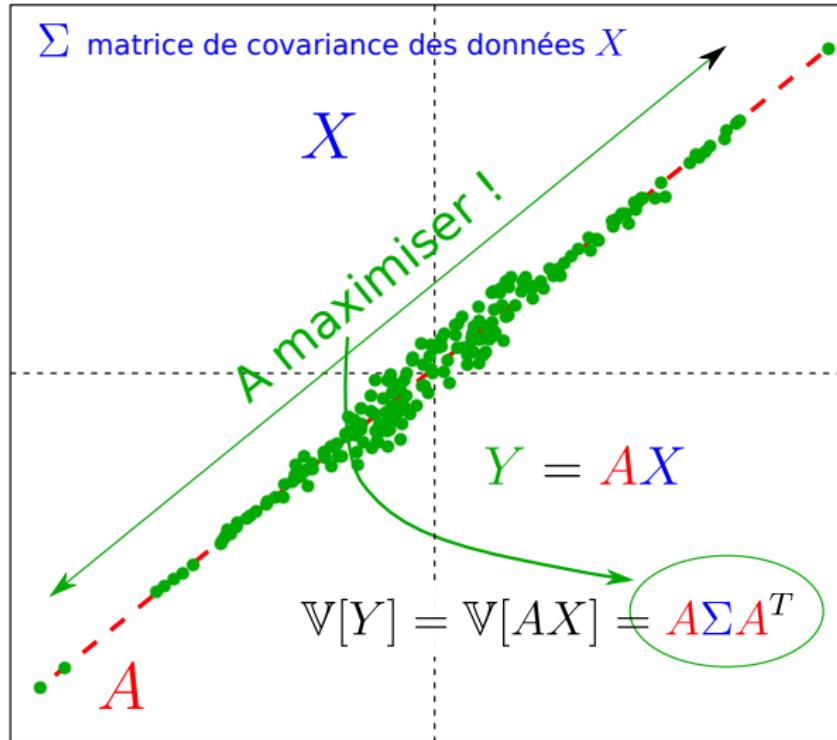
$$\frac{\partial \mathcal{L}}{\partial A} = 2\Sigma A^T - 2\lambda A^T = 0$$

Une démonstration de la TKL en 3 lignes...



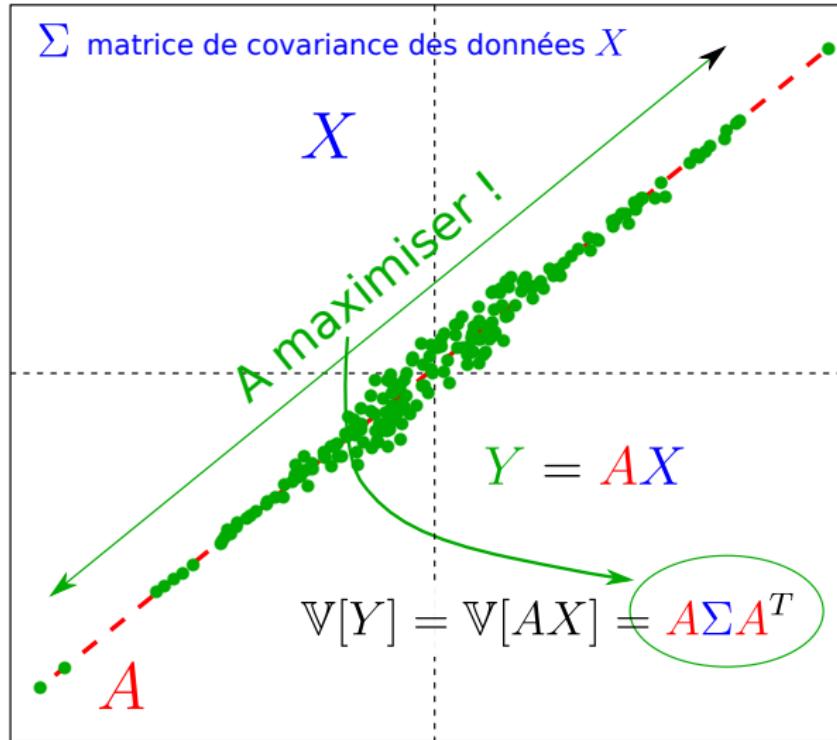
$$\Sigma A^T = \lambda A^T$$

Une démonstration de la TKL en 3 lignes...



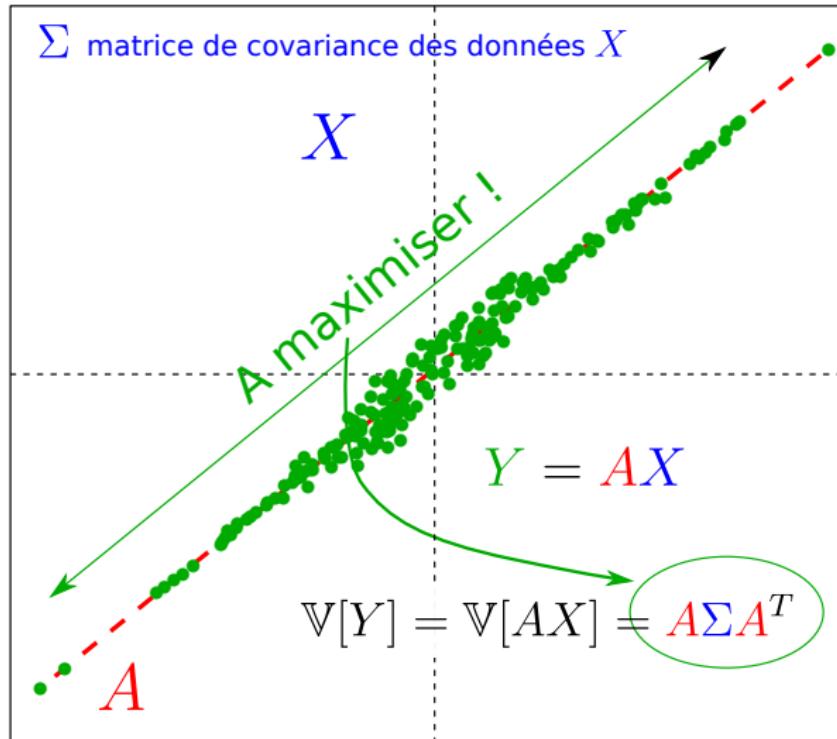
A^T est un vecteur propre de Σ !

Une démonstration de la TKL en 3 lignes...



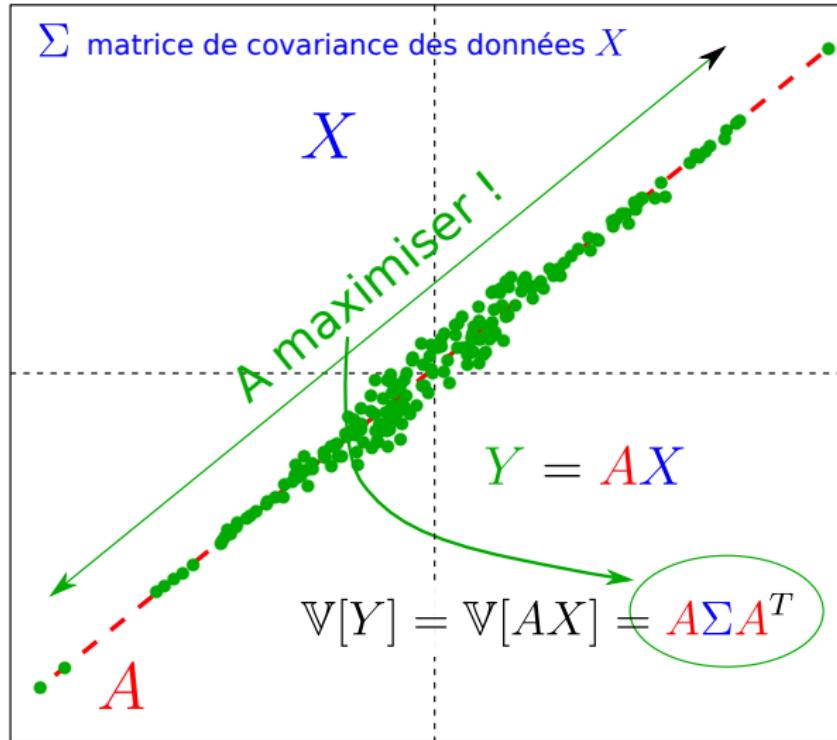
A^T est un vecteur propre de Σ ! \rightarrow Lequel ! ?

Une démonstration de la TKL en 3 lignes...



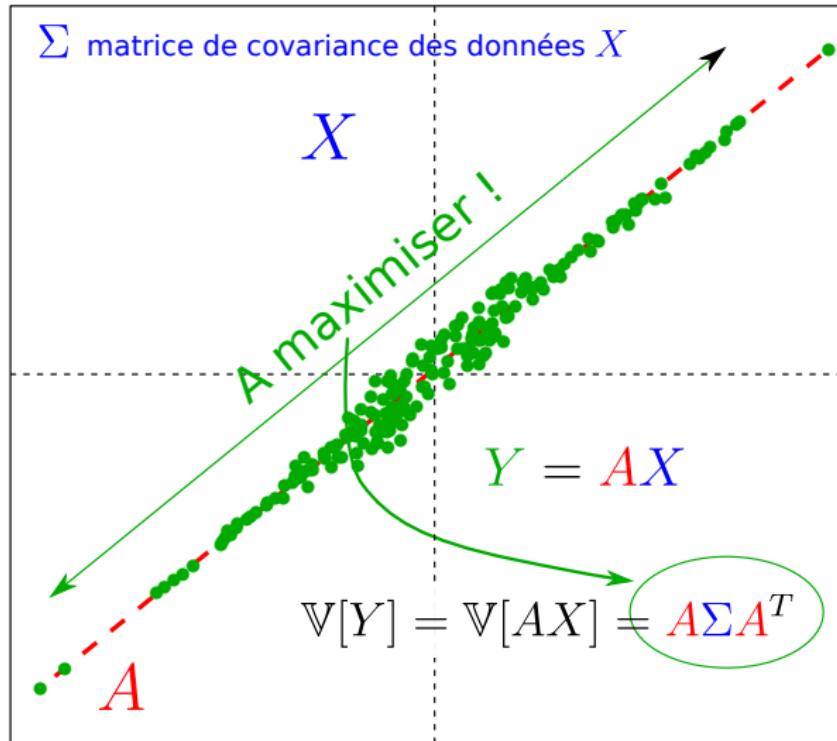
$$\Sigma A^T = \lambda A^T \Rightarrow \mathbb{V}[Y] = A\Sigma A^T = \lambda AA^T = \lambda ||A||^2 = \lambda$$

Une démonstration de la TKL en 3 lignes...



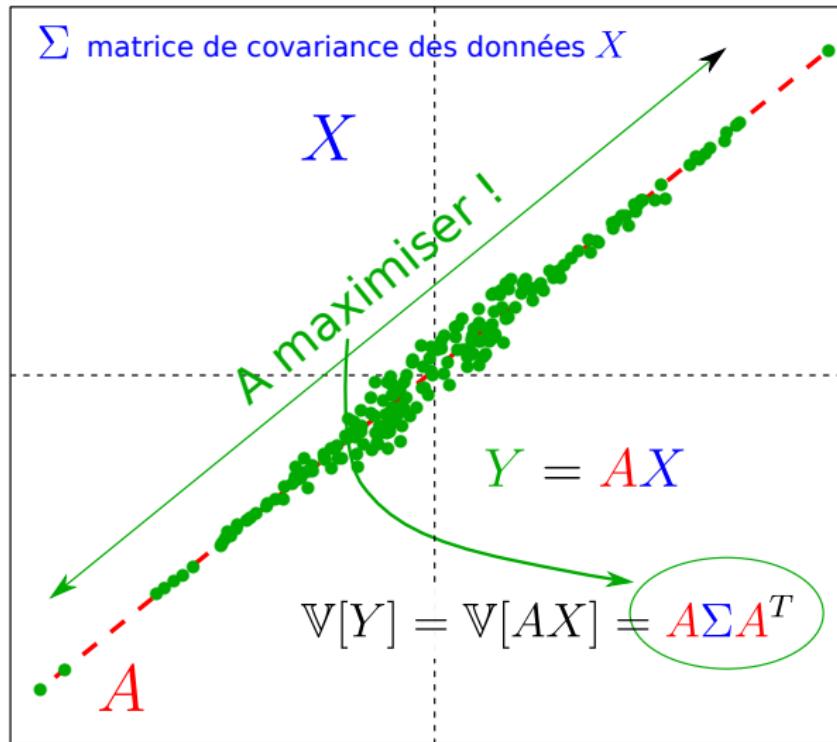
A^T est donc le vecteur propre de Σ correspondant à la plus grande valeur propre !

Une démonstration de la TKL en 3 lignes...



Récurrentivement, on continue le travail sur la projection Y pour trouver la seconde composante...

Une démonstration de la TKL en 3 lignes...



Les **composantes principales** de X sont donc les vecteurs propres de sa matrice de covariance Σ .

Au menu du jour...

- Indicateurs de performances (classifieur binaire)
- Robustesse des indicateurs
- Arbres de décision & Forêts aléatoires
- Apprentissage de données fonctionnelles
- Éléments d'apprentissage non-supervisé

Rappels : étant donné un vecteur $\mu \in \mathbb{R}^n$ et une matrice définie-positive $\Sigma \in \mathbb{R}^{n \times n}$, on appelle ***loi normale multivariée*** (de dimension n) la loi de probabilité de densité :

$$\pi(\mathbf{x}; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp \left(-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right)$$

Rappels : étant donné un vecteur $\mu \in \mathbb{R}^n$ et une matrice définie-positive $\Sigma \in \mathbb{R}^{n \times n}$, on appelle **loi normale multivariée** (de dimension n) la loi de probabilité de densité :

$$\pi(\mathbf{x}; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp \left(-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right)$$

Ayant à disposition un échantillon de réalisations de la loi π , la moyenne et la covariance empiriques sont des estimateurs optimaux (au sens des moindres carrés, ou au sens du maximum de vraisemblance) respectivement, des paramètres μ et Σ .

Mélange gaussien

Rappels : étant donné un vecteur $\mu \in \mathbb{R}^n$ et une matrice définie-positive $\Sigma \in \mathbb{R}^{n \times n}$, on appelle **loi normale multivariée** (de dimension n) la loi de probabilité de densité :

$$\pi(\mathbf{x}; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)\right)$$

Ayant à disposition un échantillon de réalisations de la loi π , la moyenne et la covariance empiriques sont des estimateurs optimaux (au sens des moindres carrés, ou au sens du maximum de vraisemblance) respectivement, des paramètres μ et Σ .

Connaissant plusieurs distributions $\pi(\cdot; \mu_i, \Sigma_i)$, on peut affecter un nouveau point \mathbf{x} au prorata des probabilités $\pi(\mathbf{x}; \mu_i, \Sigma_i)$.

Expectation-Maximization (EM)

Principe : on initialise l'algorithme en tirant k vecteurs moyennes (centres) et k matrices de covariance (dispersions).

Expectation-Maximization (EM)

Principe : on **initialise** l'algorithme en tirant k vecteurs moyennes (centres) et k matrices de covariance (dispersions).

1. On **affecte** chaque point \mathbf{x}_j des données à chaque groupe $i \in \{1, 2, \dots, k\}$ au prorata des probabilités $w_{ij} = \pi(\mathbf{x}; \mu_i, \Sigma_i)$.

Expectation-Maximization (EM)

Principe : on **initialise** l'algorithme en tirant k vecteurs moyennes (centres) et k matrices de covariance (dispersions).

1. On **affecte** chaque point \mathbf{x}_j des données à chaque groupe $i \in \{1, 2, \dots, k\}$ au prorata des probabilités $w_{ij} = \pi(\mathbf{x}; \mu_i, \Sigma_i)$.
2. On **recalcule** les paramètres μ_i et Σ_i de chaque centre, en utilisant toutes les points de données $\pi(\mathbf{x})$, pondérés par leur appartenance au groupe i (déterminée au point 1.).

Expectation-Maximization (EM)

Principe : on **initialise** l'algorithme en tirant k vecteurs moyennes (centres) et k matrices de covariance (dispersions).

1. On **affecte** chaque point \mathbf{x}_j des données à chaque groupe $i \in \{1, 2, \dots, k\}$ au prorata des probabilités $w_{ij} = \pi(\mathbf{x}; \mu_i, \Sigma_i)$.
2. On **recalcule** les paramètres μ_i et Σ_i de chaque centre, en utilisant toutes les points de données $\pi(\mathbf{x})$, pondérés par leur appartenance au groupe i (déterminée au point 1.).

E.g. (pour la moyenne) :

$$\mu_i^+ = \sum_{j=1}^n w_{ij} \mathbf{x}_j \quad \forall i \in \{1, 2, \dots, k\}$$

Expectation-Maximization (EM)

Principe : on **initialise** l'algorithme en tirant k vecteurs moyennes (centres) et k matrices de covariance (dispersions).

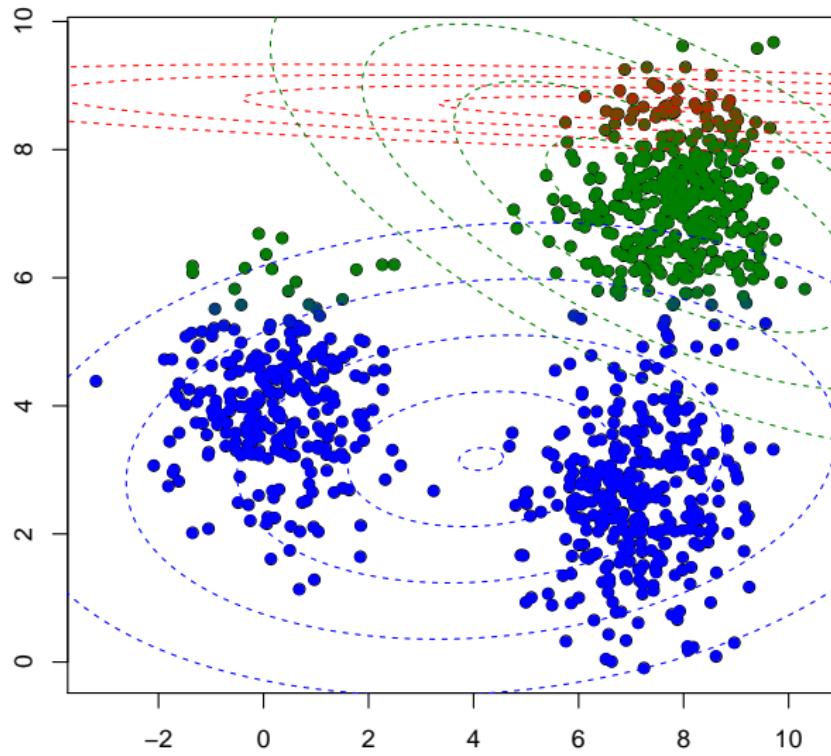
1. On **affecte** chaque point \mathbf{x}_j des données à chaque groupe $i \in \{1, 2, \dots, k\}$ au prorata des probabilités $w_{ij} = \pi(\mathbf{x}; \mu_i, \Sigma_i)$.
2. On **recalcule** les paramètres μ_i et Σ_i de chaque centre, en utilisant toutes les points de données $\pi(\mathbf{x})$, pondérés par leur appartenance au groupe i (déterminée au point 1.).

E.g. (pour la moyenne) :

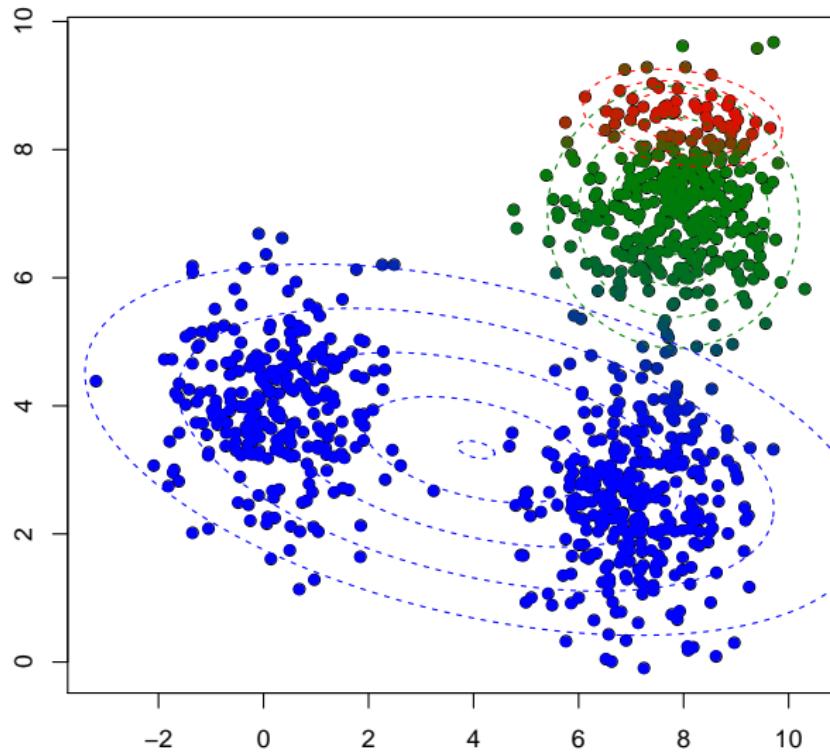
$$\mu_i^+ = \sum_{j=1}^n w_{ij} \mathbf{x}_j \quad \forall i \in \{1, 2, \dots, k\}$$

On itère entre les étapes 1. et 2. jusqu'à convergence (garantie).

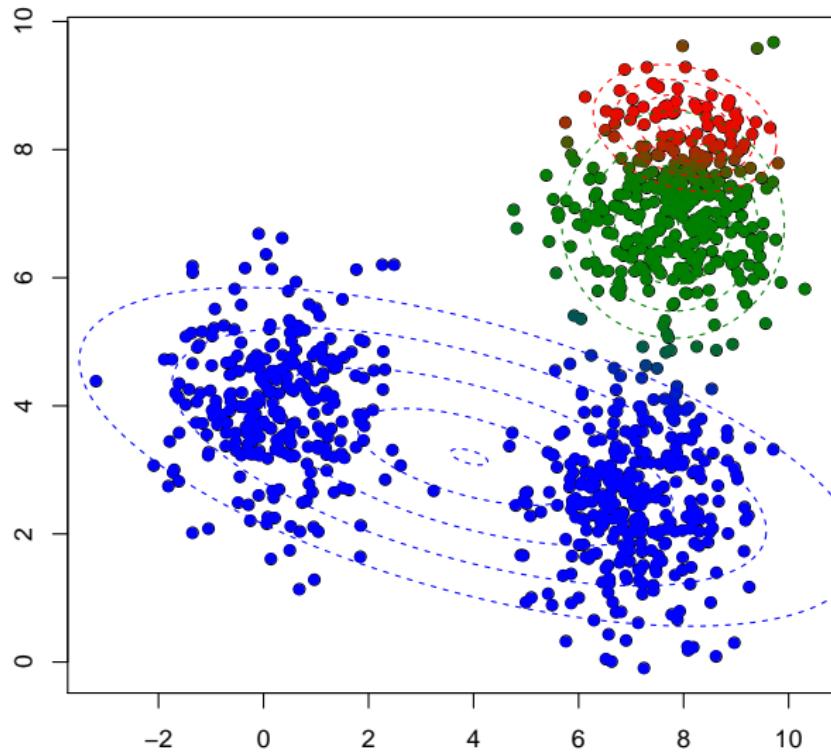
Expectation-Maximization (EM)



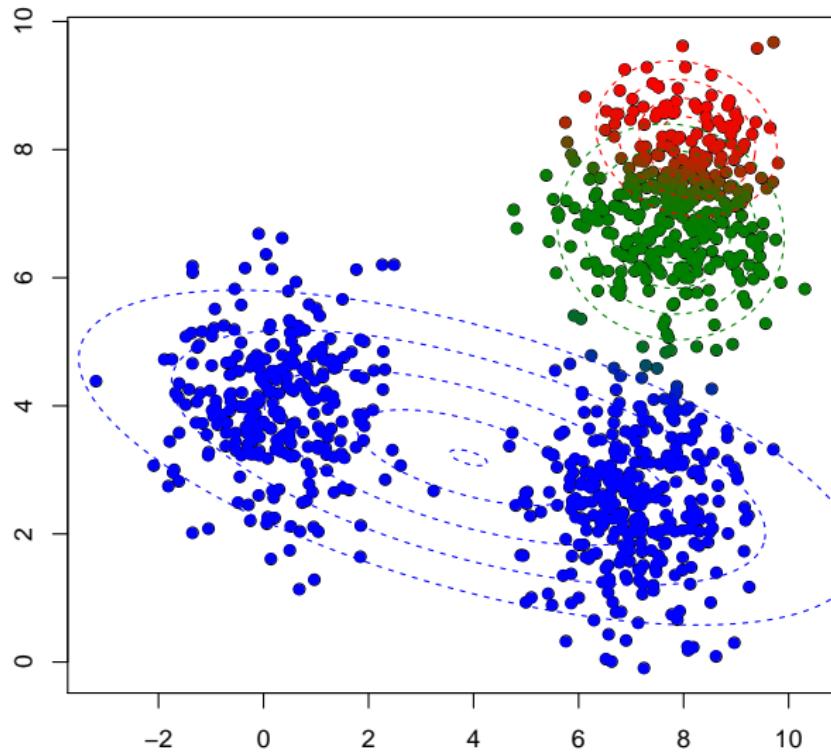
Expectation-Maximization (EM)



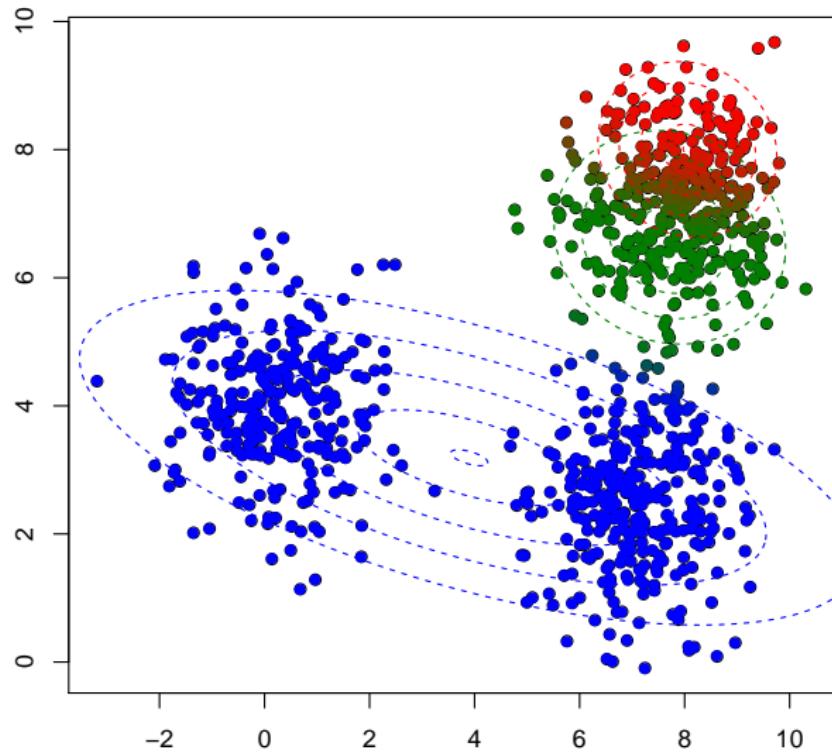
Expectation-Maximization (EM)



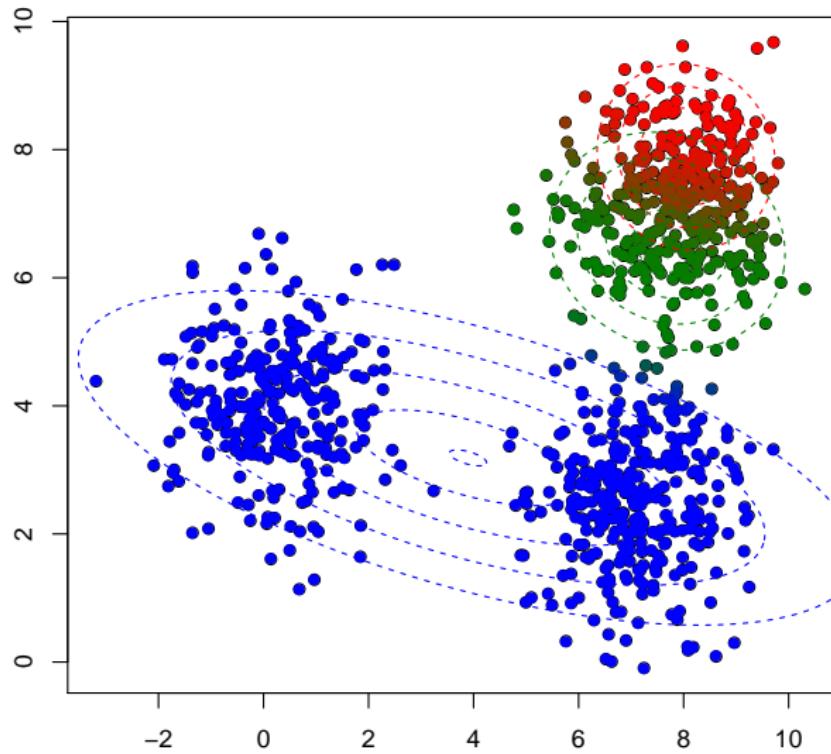
Expectation-Maximization (EM)



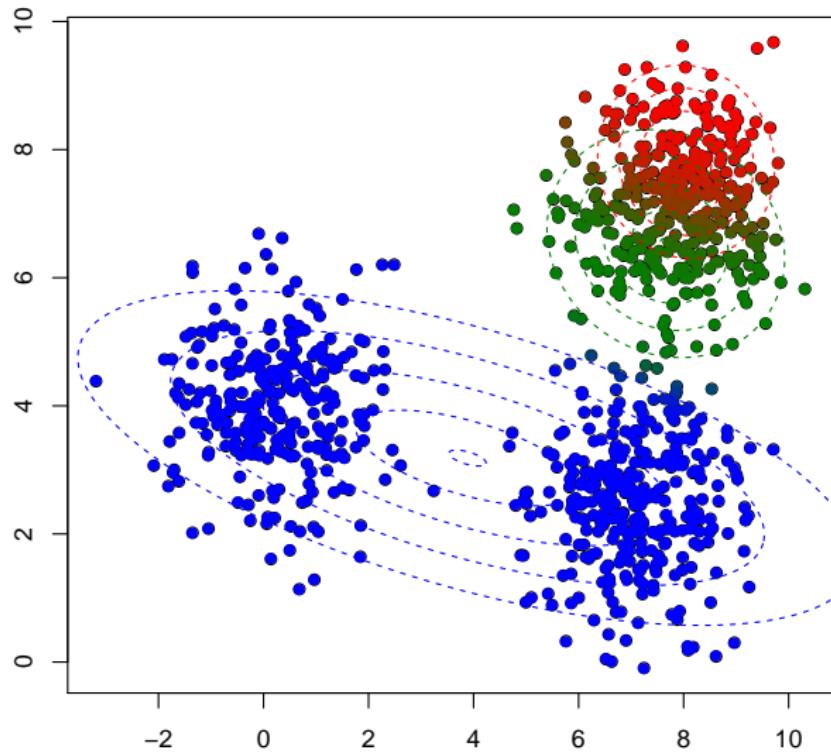
Expectation-Maximization (EM)



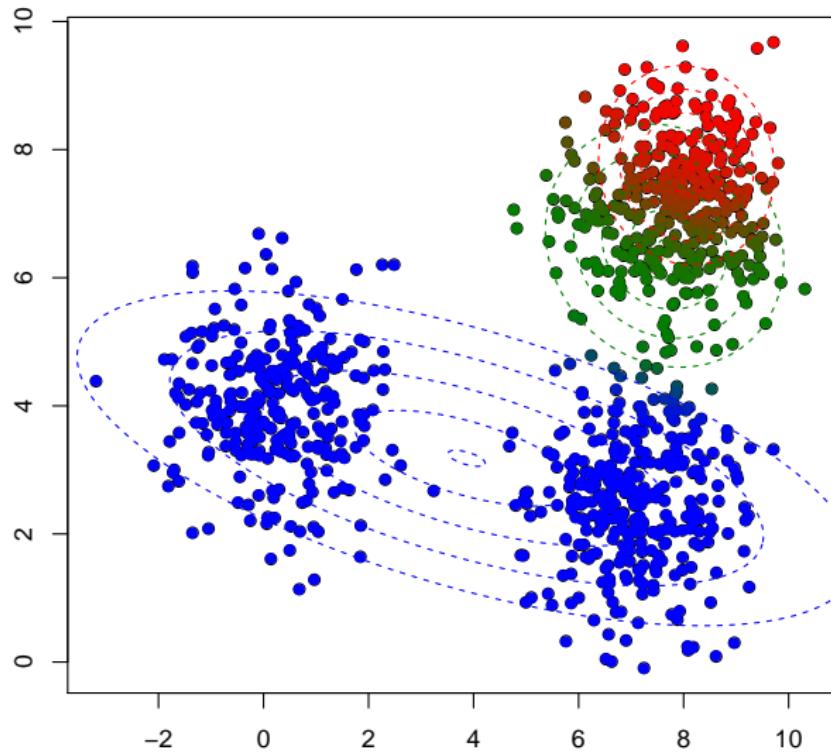
Expectation-Maximization (EM)



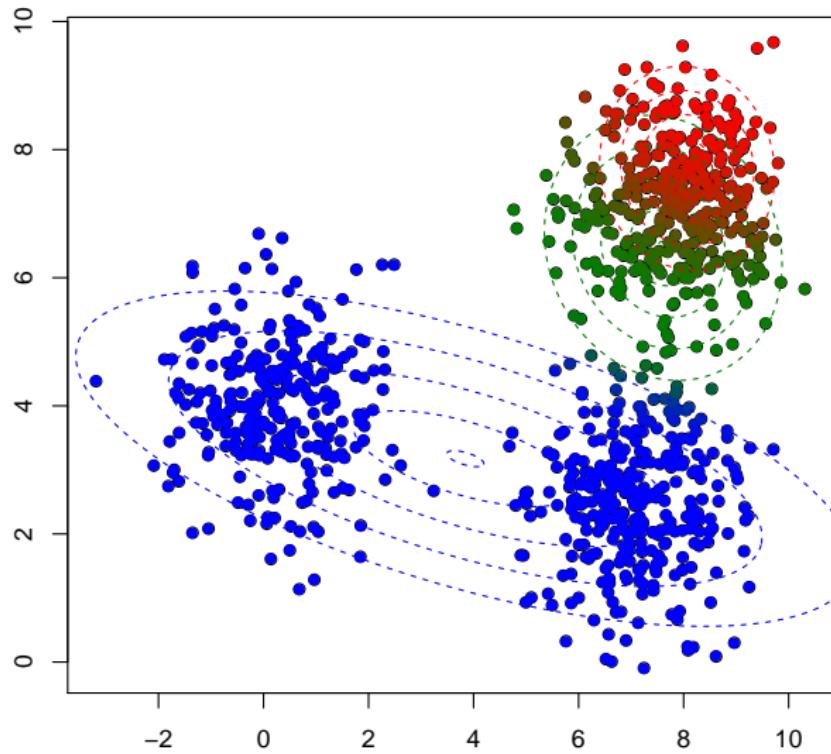
Expectation-Maximization (EM)



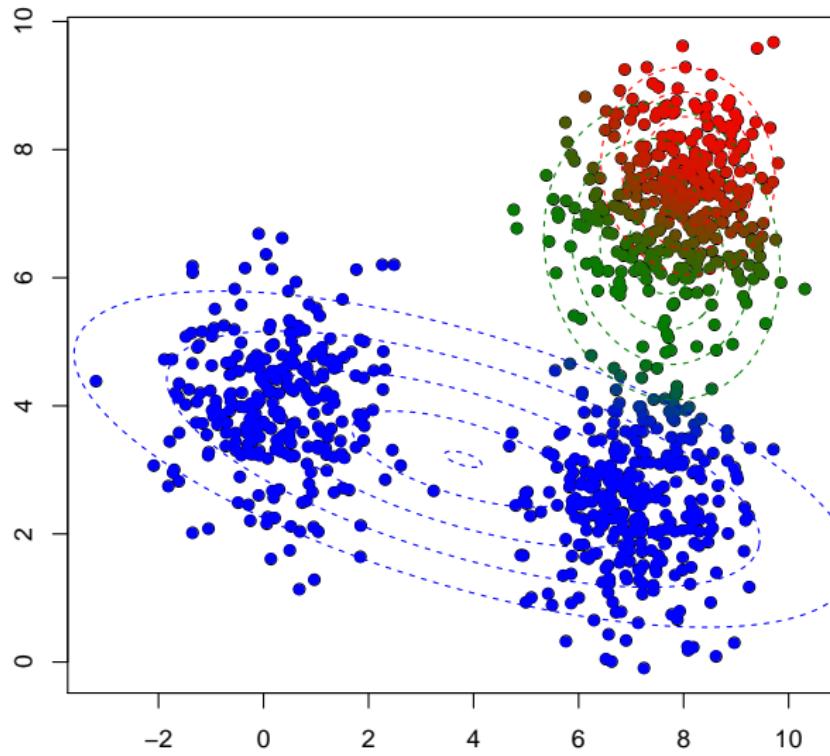
Expectation-Maximization (EM)



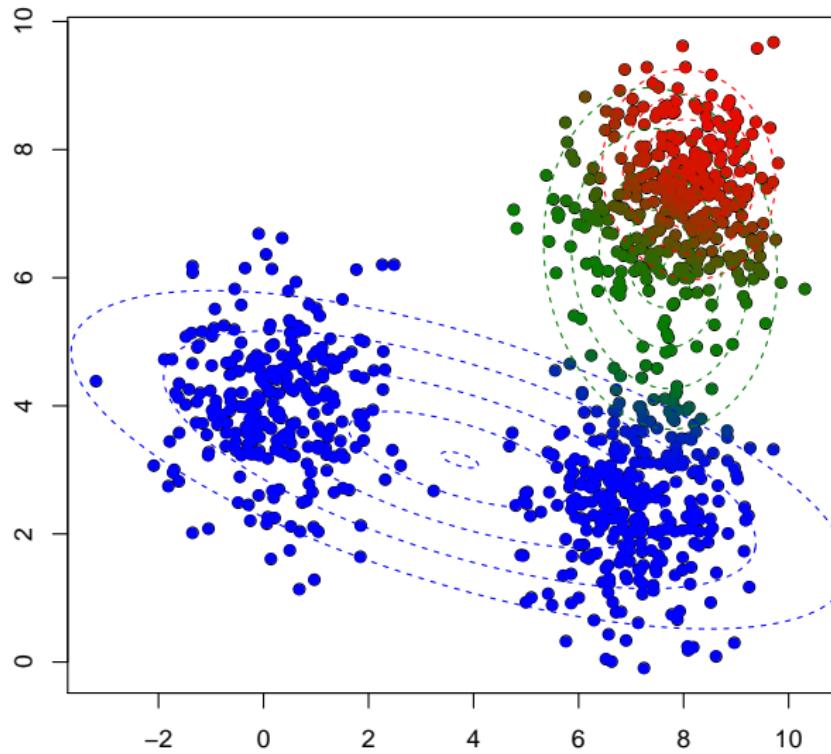
Expectation-Maximization (EM)



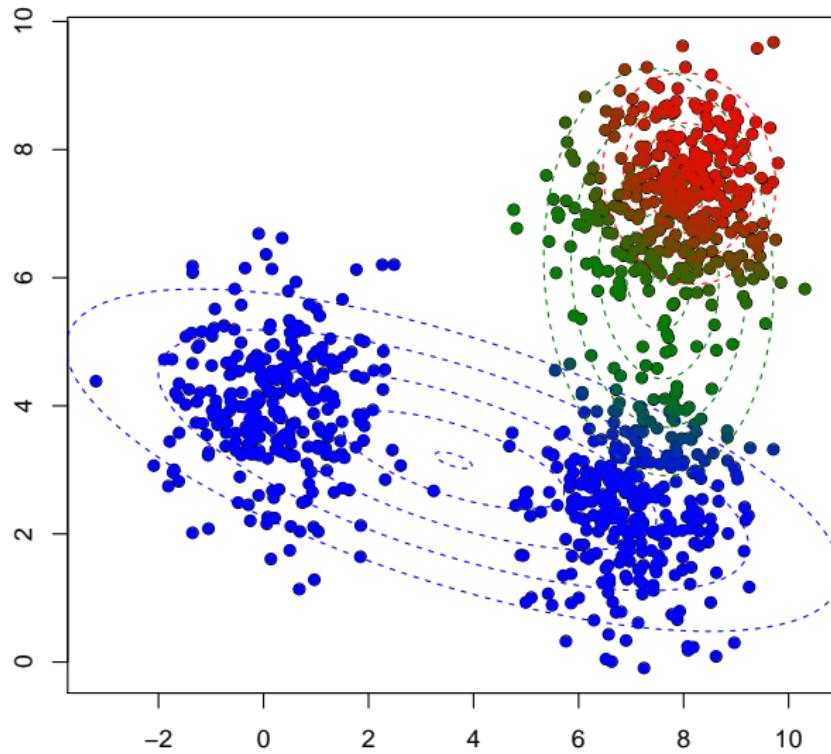
Expectation-Maximization (EM)



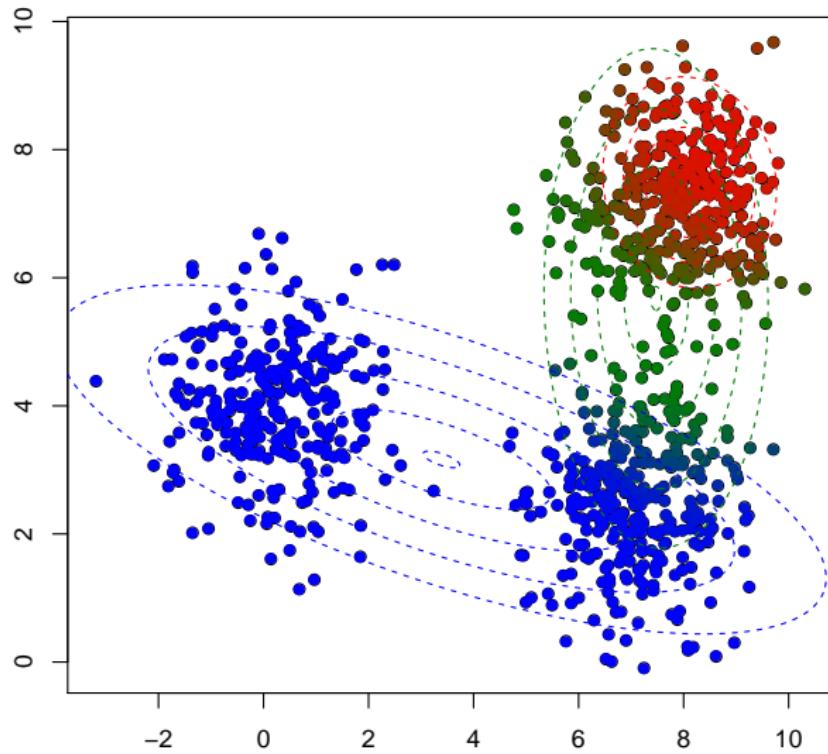
Expectation-Maximization (EM)



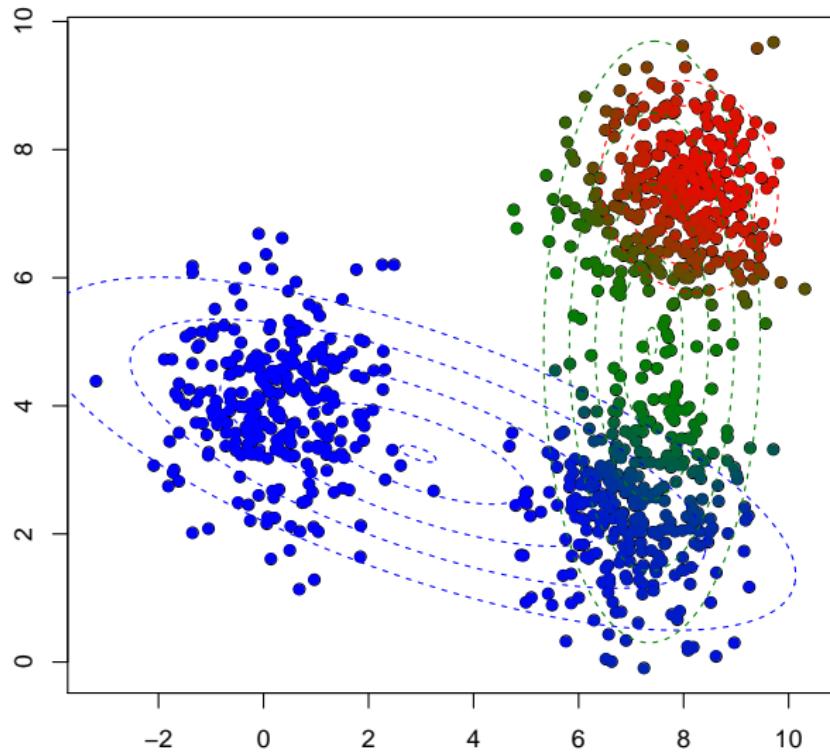
Expectation-Maximization (EM)



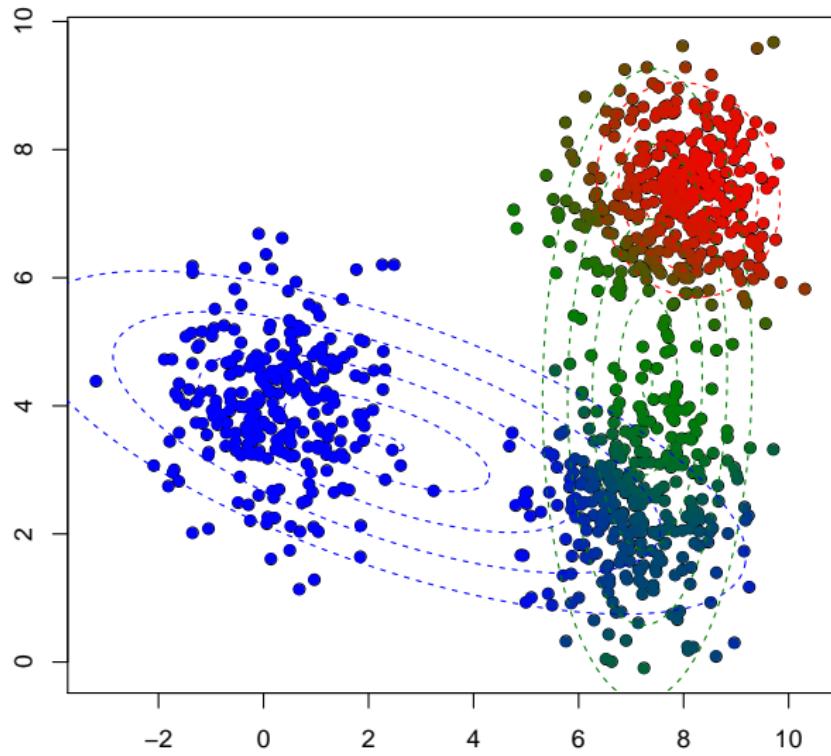
Expectation-Maximization (EM)



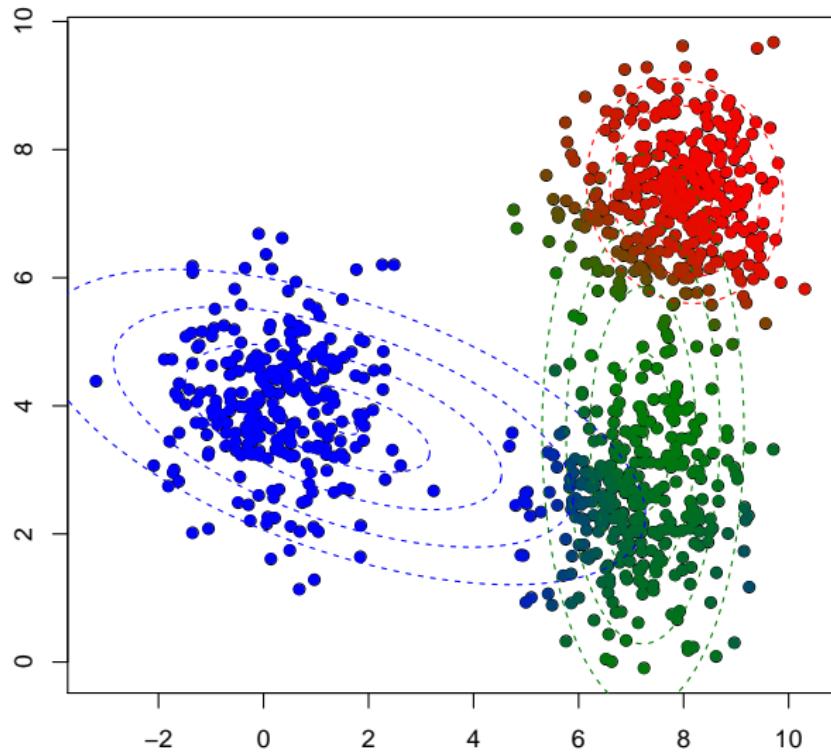
Expectation-Maximization (EM)



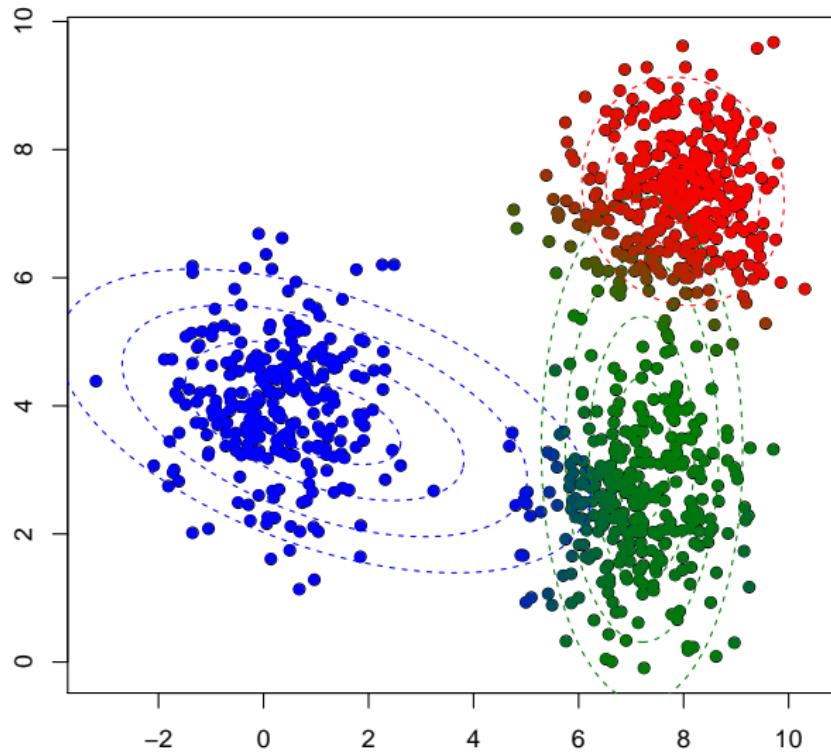
Expectation-Maximization (EM)



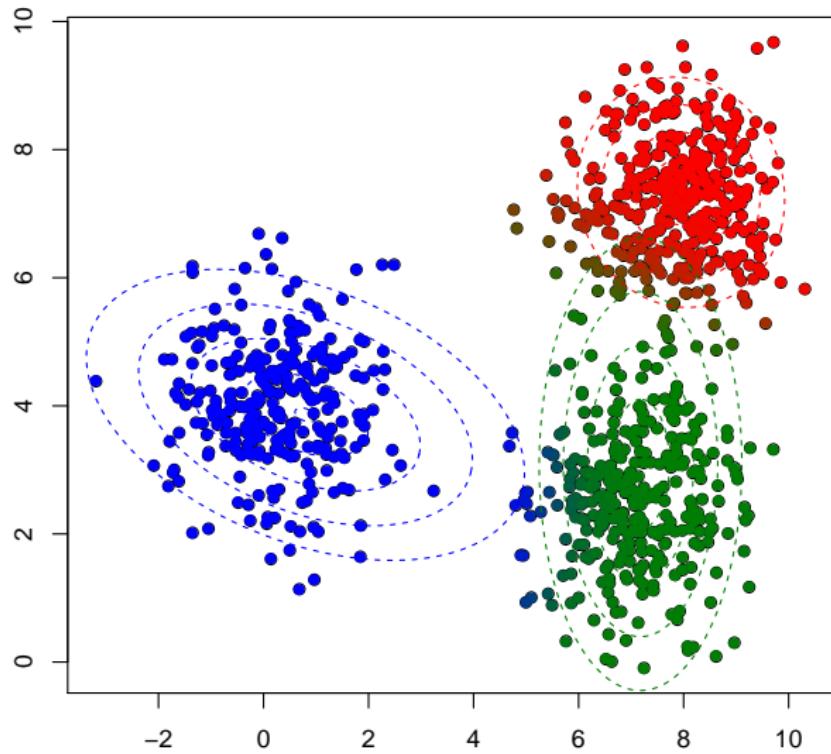
Expectation-Maximization (EM)



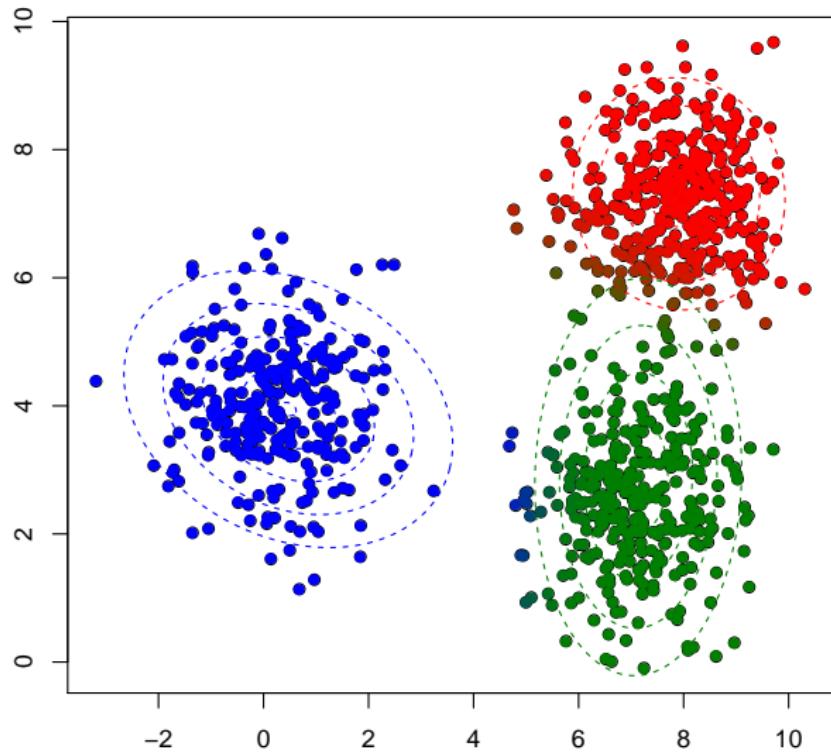
Expectation-Maximization (EM)



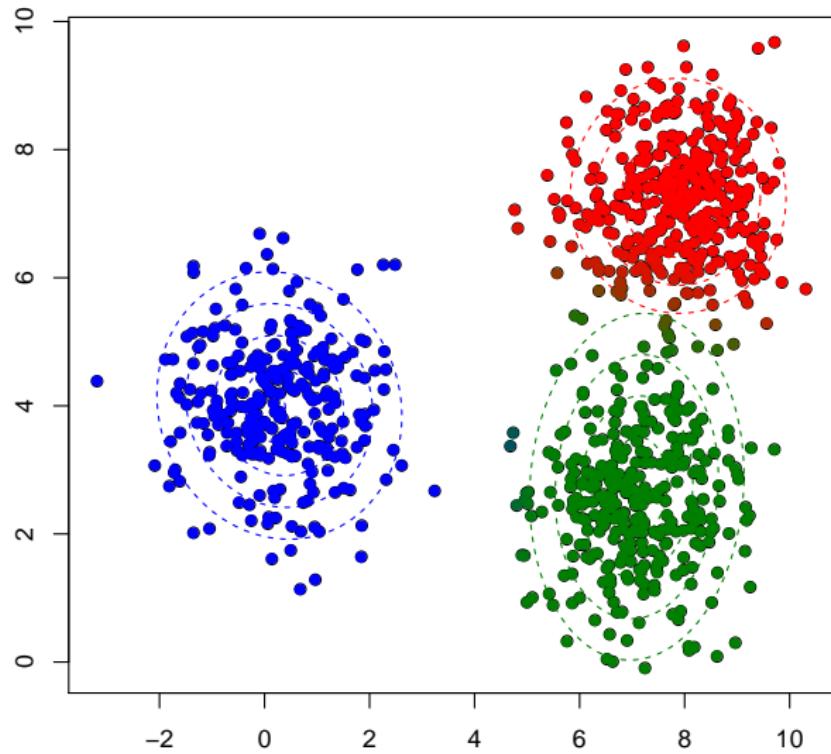
Expectation-Maximization (EM)



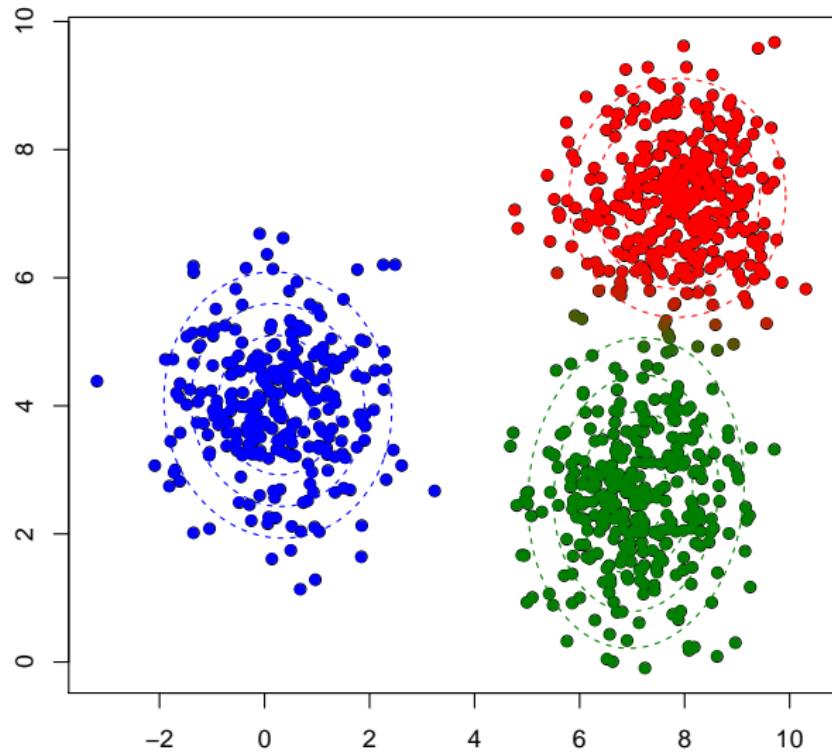
Expectation-Maximization (EM)



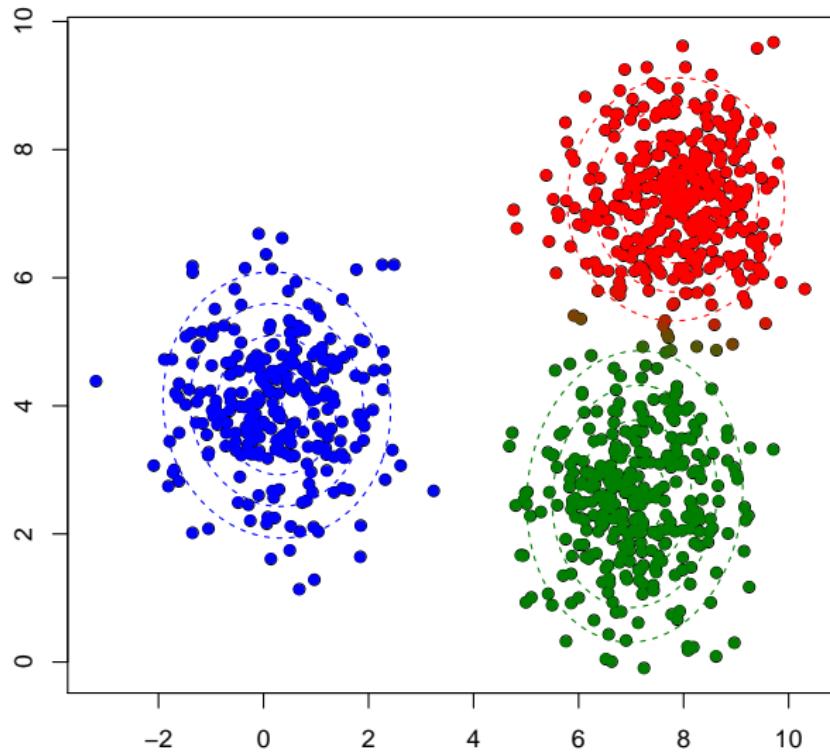
Expectation-Maximization (EM)



Expectation-Maximization (EM)



Expectation-Maximization (EM)



Expectation-Maximization (EM)

