Quelques références bibliographiques

- G. Dreyfus, J. Martinez, M. Samuelides, M. Gordon, F. Badran, S. Thiria. Apprentissage statistique : Réseaux de neurones - Cartes topologiques - Machines à vecteurs supports. Éditions Eyrolles, 2008, 3ème édition.
- B. Schölkopf, A. Smola. *Learning with Kernels*. MIT Press, 2002. [4]
- I. Goodfellow, Y. Bengio, A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`. [3]

D'autres références vous seront suggérées dans les différents chapitres du cours

Plan du cours

Plan du cours

Modèle décisionnel

- Observations décrites par les valeurs prises par un ensemble de variables

→ Objectif : prédire, pour chaque donnée, la valeur d'une variable (expliquée ou « dépendante » ou « de sortie ») à partir des valeurs des autres variables (explicatives ou « d'entrée »)

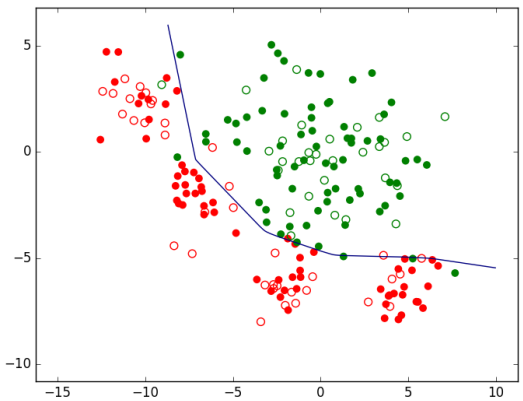
- Exemples
  1. Une région d'une image représente un visage ou non ?
  2. Les symptômes correspondent à la maladie A ou B ou C ou aucune ?
  3. Quel est le volume d'algues vertes attendu en mai sur les plages de la commune ?
  4. Quel sera le débit de la Loire à Tours dans 48h ?
  5. Quelle est l'entité nommée dans « La Maison Blanche a démenti ces informations. » ?
  6. Quelle est la région d'image correspondant aux pantalons ?

Types de problèmes de décision

1. Classement (ou discrimination) : la variable expliquée est une variable nominale, chaque observation possède une modalité (appelée en général classe)
2. Régression : la variable expliquée est une variable quantitative (domaine $\subset \mathbb{R}$)
3. Prédiction structurée : la variable expliquée prend des valeurs dans un domaine de données structurées (les relations entre parties comptent)

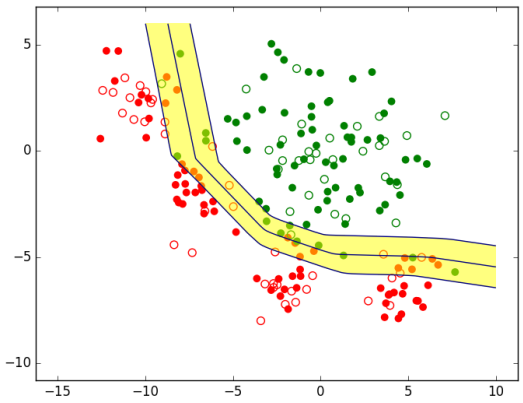
Qu'est-ce qu'un modèle ?

- Modèle = règle de décision
- Exemple : frontière de discrimination pour problème de classement à 2 classes



Qu'est-ce qu'un modèle ?

- Modèle = règle de décision
- Exemple : frontière de discrimination pour problème de classement à 2 classes

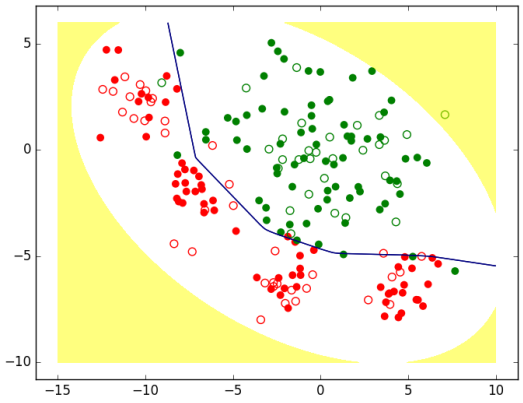


- Éventuellement complété par des critères de rejet (refus d'affectation)
  1. Refus de classer les données trop proches de la frontière (rejet d'ambiguïté)

Modélisation décisionnelle | Types de problèmes de décision | 11 / 56

Qu'est-ce qu'un modèle ?

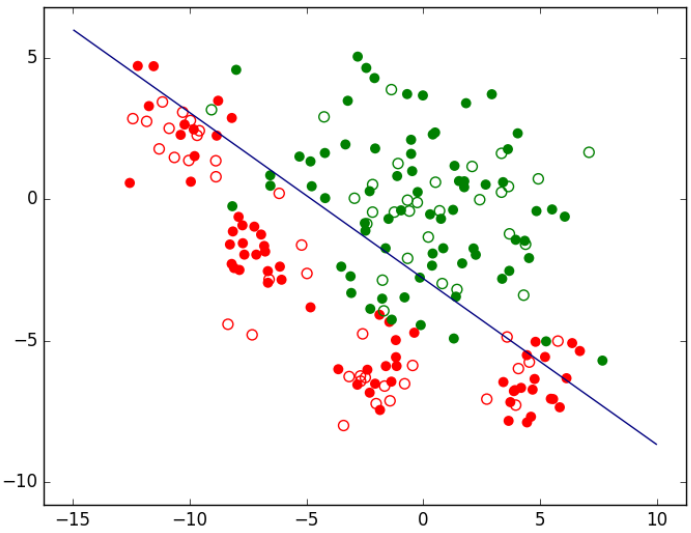
- Modèle = règle de décision
- Exemple : frontière de discrimination pour problème de classement à 2 classes



- Éventuellement complété par des critères de rejet (refus d'affectation)
  2. Refus de classer les données trop éloignées des données connues (rejet de non représentativité)

Classement

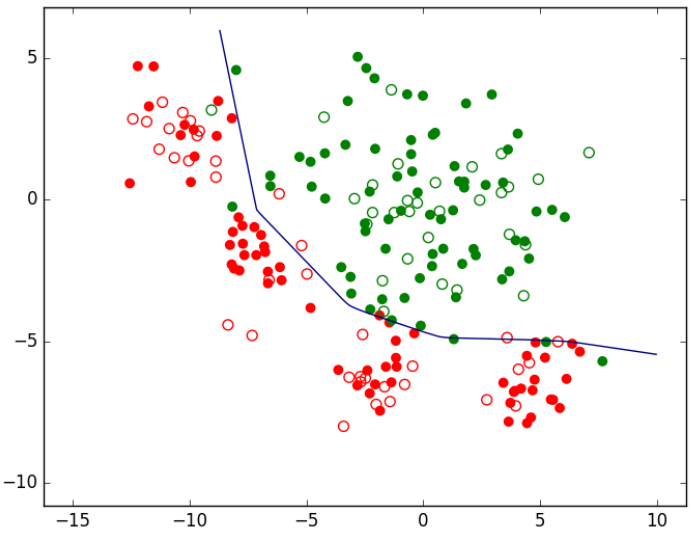
- Modèle : règle de classement, par ex. frontière de discrimination (trait bleu foncé)
- Exemple : (2 var. explicatives pour chaque observation : abscisse X et ordonnée Y)



Modélisation décisionnelle | Types de problèmes de décision | 12 / 56

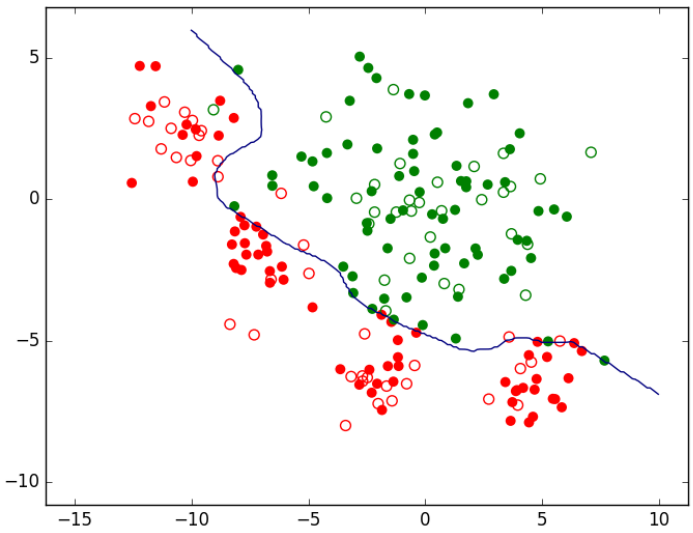
Classement

- Modèle : règle de classement, par ex. frontière de discrimination (trait bleu foncé)
- Exemple : (2 var. explicatives pour chaque observation : abscisse X et ordonnée Y)



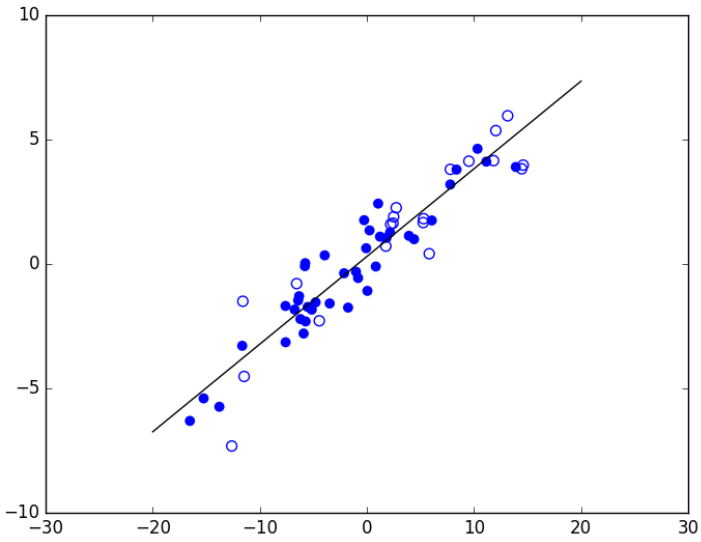
Classement

- Modèle : règle de classement, par ex. frontière de discrimination (trait bleu foncé)
- Exemple : (2 var. explicatives pour chaque observation : abscisse X et ordonnée Y)



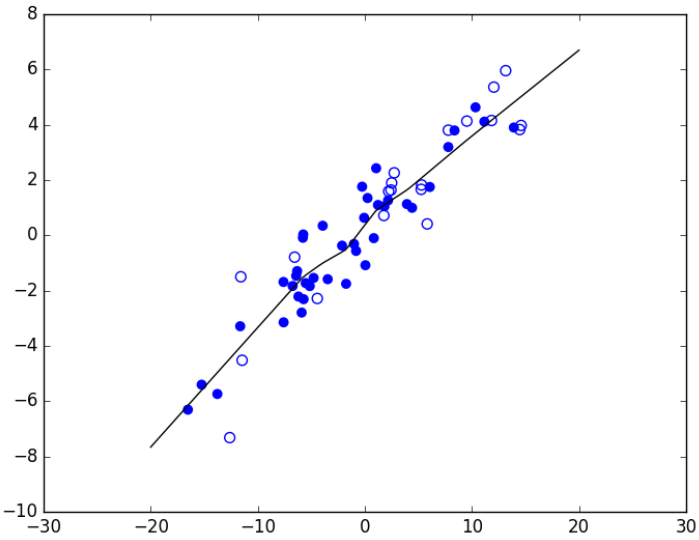
Régression

- Modèle : règle de prédiction (trait noir dans la figure)
  - Par ex. $y = ax + b$ pour modèle linéaire
- Exemple : (variable explicative X en abscisse, variable expliquée Y en ordonnée)



Régression

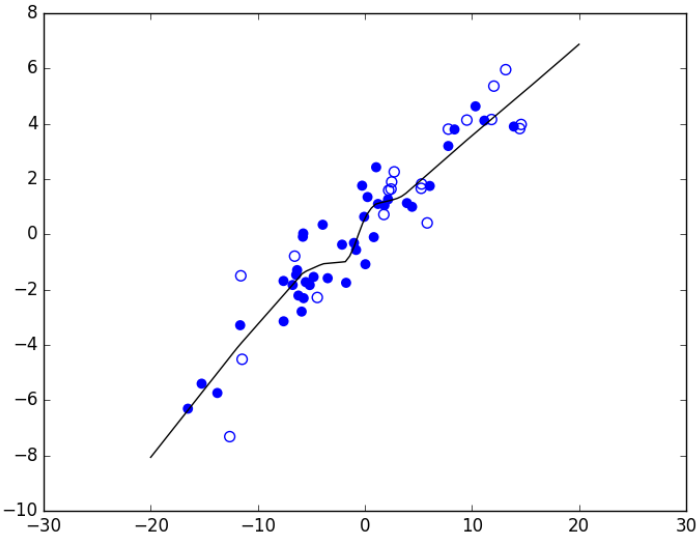
- Modèle : règle de prédiction (trait noir dans la figure)
- Exemple : (variable explicative X en abscisse, variable expliquée Y en ordonnée)



Modélisation décisionnelle | Types de problèmes de décision | 13 / 56

Régression

- Modèle : règle de prédiction (trait noir dans la figure)
- Exemple : (variable explicative X en abscisse, variable expliquée Y en ordonnée)



Prédiction structurée

- Modèle : règle de prédiction
- Exemples :
  1. Déterminer que l'entité nommée de la phrase « La Maison Blanche a démenti ces informations. » est La Maison Blanche
     - Les classements des mots composant l'entité nommée ne sont pas indépendants

Modélisation décisionnelle | Types de problèmes de décision | 14 / 56

Prédiction structurée

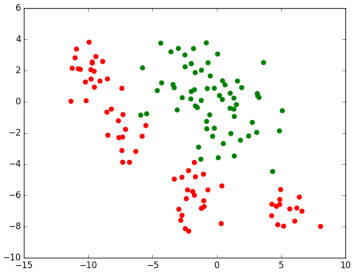
- Modèle : règle de prédiction
- Exemples :
  1. Déterminer que l'entité nommée de la phrase « La Maison Blanche a démenti ces informations. » est La Maison Blanche
     - Les classements des mots composant l'entité nommée ne sont pas indépendants
  2. Délimiter la région correspondant aux pantalons dans l'image [5]

- Les affectations des pixels composant la région ne sont pas indépendantes

Comment obtenir un modèle décisionnel

1. Construction analytique, à partir d'une parfaite connaissance du phénomène
   - Exemples :
     - Temps de vol ← distance et vitesse
     - Concentration de produit de réaction ← concentration de réactif et température
   - Néglige souvent l'impact de variables non contrôlables !
2. A partir de données : ensemble d'observations pour lesquelles les valeurs des variables explicatives et des variables expliquées sont en général connues

   → Apprentissage supervisé : à partir d'observations pour lesquelles les valeurs des variables explicatives et de la variable expliquée sont connues

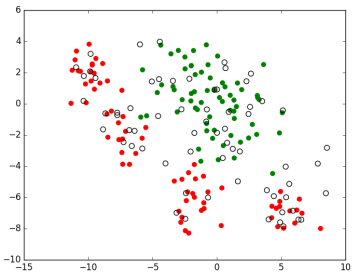


Modélisation décisionnelle | Modélisation à partir de données 15 / 56

Comment obtenir un modèle décisionnel

1. Construction analytique, à partir d'une parfaite connaissance du phénomène
   - Exemples :
     - Temps de vol ← distance et vitesse
     - Concentration de produit de réaction ← concentration de réactif et température
   - Néglige souvent l'impact de variables non contrôlables !
2. A partir de données : ensemble d'observations pour lesquelles les valeurs des variables explicatives et des variables expliquées sont en général connues

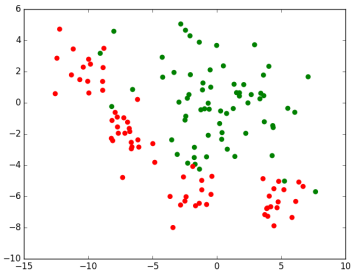
   → Apprentissage supervisé : à partir d'observations pour lesquelles les valeurs des variables explicatives et de la variable expliquée sont connues



   - Apprentissage semi-supervisé (voir [2]) : tient compte aussi des observations pour lesquelles les valeurs de la variable expliquée sont inconnues

Apprentissage et généralisation

- (Information de) Supervision = valeur de la variable expliquée
- Modélisation à partir de données (observations) d'apprentissage, qui disposent de l'information de supervision
- Choix famille paramétrique, puis optimisation des paramètres $\rightarrow$ modèle
- Erreur du modèle sur ces données = erreur d'apprentissage ou risque empirique

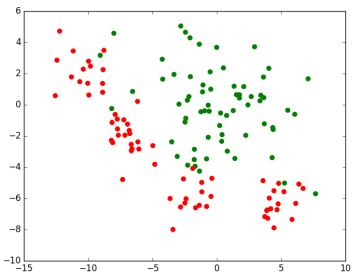


données

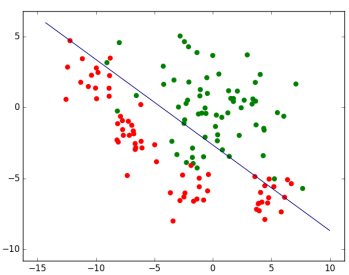
Modélisation décisionnelle | Modélisation à partir de données 16 / 56

Apprentissage et généralisation

- (Information de) Supervision = valeur de la variable expliquée
- Modélisation à partir de données (observations) d'apprentissage, qui disposent de l'information de supervision
- Choix famille paramétrique, puis optimisation des paramètres $\rightarrow$ modèle
- Erreur du modèle sur ces données = erreur d'apprentissage ou risque empirique



données

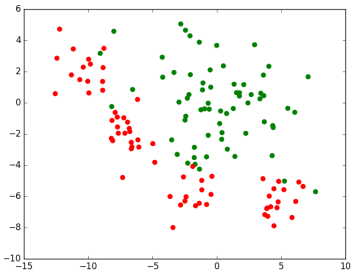


AFD

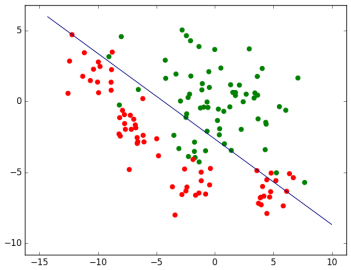
erreur = 12%

Apprentissage et généralisation

- (Information de) Supervision = valeur de la variable expliquée
- Modélisation à partir de données (observations) d'apprentissage, qui disposent de l'information de supervision
- Choix famille paramétrique, puis optimisation des paramètres → modèle
- Erreur du modèle sur ces données = erreur d'apprentissage ou risque empirique

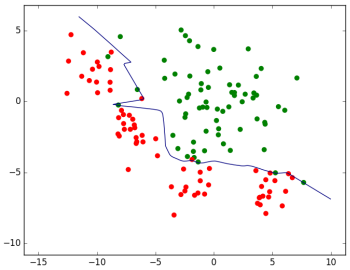


données



AFD

erreur = 12%

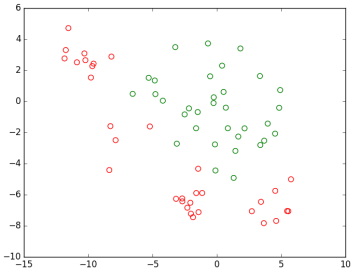


PMC $\alpha = 10^{-5}$

erreur = 2,3%

Apprentissage et généralisation (2)

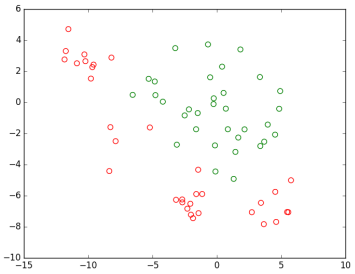
- Le modèle permet de prendre des décisions pour de futures (nouvelles) données
- Erreur du modèle sur ces futures données = erreur de généralisation ou risque espéré



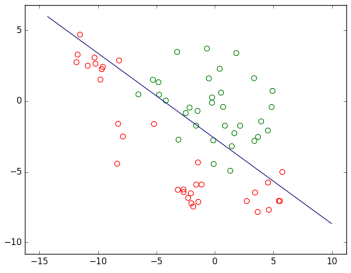
données

Apprentissage et généralisation (2)

- Le modèle permet de prendre des décisions pour de futures (nouvelles) données
- Erreur du modèle sur ces futures données = erreur de généralisation ou risque espéré



données



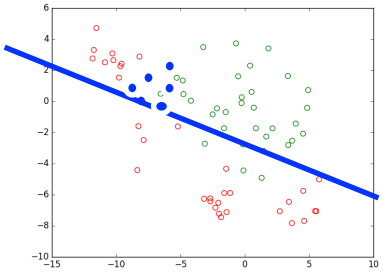
AFD

erreur = 14%

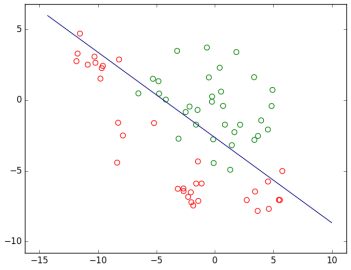
Modélisation décisionnelle | Modélisation à partir de données 17 / 56

Apprentissage et généralisation (2)

- Le modèle permet de prendre des décisions pour de futures (nouvelles) données
- Erreur du modèle sur ces futures données = erreur de généralisation ou risque espéré

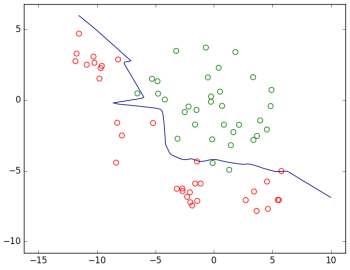


données



AFD

erreur = 14%



PMC $\alpha = 10^{-5}$

erreur = 6%

→ Objectif : avoir la meilleure généralisation (le risque espéré le plus faible)

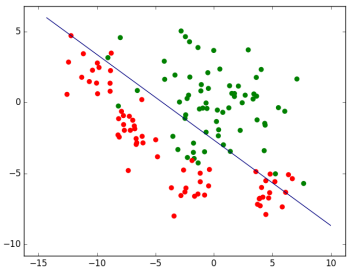
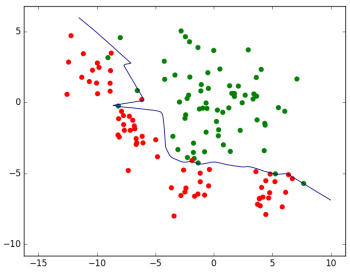
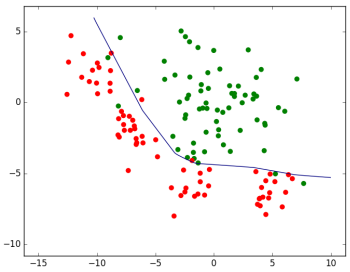
Comment trouver le modèle qui présente la meilleure généralisation ?

- Erreur d'apprentissage (facilement) mesurable car ces données sont disponibles
- Données futures inconnues $\Rightarrow$ erreur de généralisation ne peut pas être mesurée
- Hypothèse importante : la distribution des données d'apprentissage est représentative de celle des données futures !
  - Or, on constate souvent que la distribution évolue dans le temps (n'est pas stationnaire) $\Rightarrow$ il est nécessaire d'adapter régulièrement le modèle

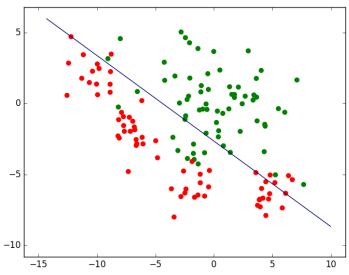
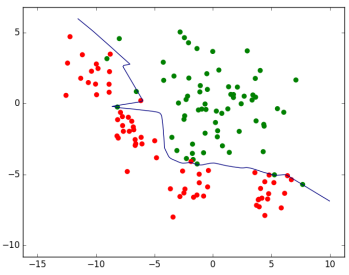
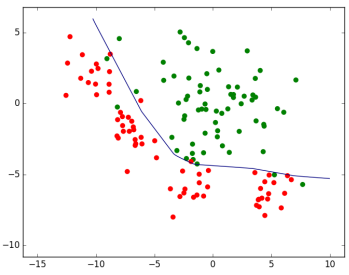
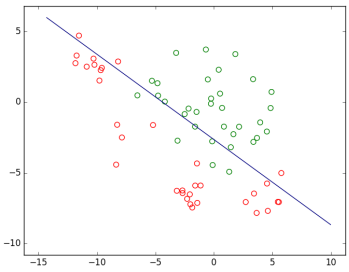
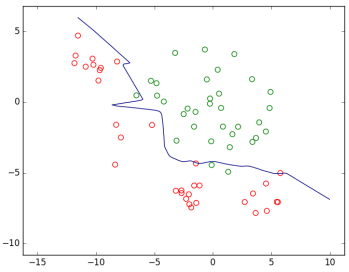
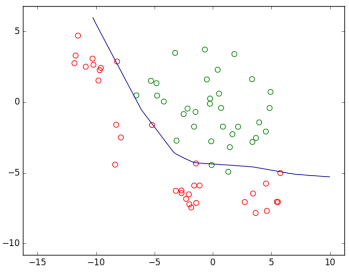
$\rightarrow$ Minimiser l'erreur d'apprentissage permet de minimiser l'erreur de généralisation ?

- Considérons des données de test, non utilisées pour l'apprentissage mais disposant de l'information de supervision
- Comparons trois modèles différents :
  - Modèle linéaire obtenu par analyse factorielle discriminante (AFD)
  - Perceptron multicouches (PMC) avec un coefficient « d'oubli » (*weight decay*) $\alpha = 10^{-5}$
  - Perceptron multicouches (PMC) avec un coefficient « d'oubli » $\alpha = 1$

Quel lien entre erreur d'apprentissage et erreur de généralisation ?

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Apprentissage			
	Err. app. 12%	2,3%	4,5%

Quel lien entre erreur d'apprentissage et erreur de généralisation ?

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Apprentissage			
	Err. app. 12%	2,3%	4,5%
Test			
	Err. test 14%	6%	4,6%

Quel lien entre erreur d'apprentissage et erreur de généralisation ? (2)

- Constats
  1. Le modèle qui a la plus faible erreur d'apprentissage n'a pas la plus faible erreur de test
     - Cela reste valable si on compare des modèles issus de la même famille, par ex. par arrêt précoce de la procédure d'optimisation
  2. L'erreur d'apprentissage est en général une estimation optimiste de l'erreur de test
  3. L'écart entre erreur d'apprentissage et erreur de test dépend de la famille de modèles

- Si on ne peut pas mesurer l'erreur de généralisation, comment l'estimer ?
  1. Par l'erreur sur des données de test, non utilisées pour l'apprentissage
     → Les observations disponibles avec information de supervision sont séparées en données d'apprentissage (→ obtenir le modèle) et données de test (→ estimer la généralisation)
  2. Grâce à une éventuelle borne supérieure sur l'écart entre erreur d'apprentissage et erreur de généralisation : erreur généralisation $\leq$ erreur apprentissage + borne
     → Lorsqu'elle existe, la borne peut être trop élevée pour être exploitable

Plan du cours

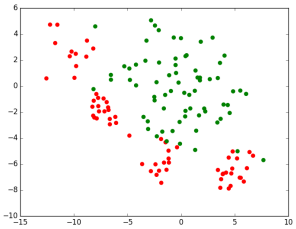
Modélisation à partir de données : étapes générales

1. Préparation des données et choix d'une fonction de perte (*loss* ou erreur)
2. Choix des familles paramétriques dans lesquelles chercher des modèles
3. Dans chaque famille, estimation du « meilleur » modèle intra-famille
4. Choix du meilleur modèle entre familles
5. Évaluation des performances de généralisation du modèle retenu

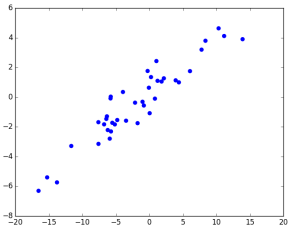
Modélisation à partir de données : un cadre plus précis

- Domaine des variables explicatives (ou espace d'entrée) : $\mathcal{X}$ (par ex. $\mathbb{R}^p$)
- Domaine de la variable expliquée (ou espace de sortie) : $\mathcal{Y}$ (par ex. $\{-1\,;1\}$, $\mathbb{R}$)
- Données à modéliser décrites par variables aléatoires $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ suivant la distribution inconnue P
- Exemples

Classement :
$\mathcal{X} \subset \mathbb{R}^2$
$\mathcal{Y} = \{c_1\,, c_2\}$



Régression :
$\mathcal{X} \subset \mathbb{R}$
$\mathcal{Y} \subset \mathbb{R}$



Modélisation à partir de données : un cadre plus précis (2)

- Observations (données) avec information de supervision : $\mathcal{D}_N = \{(\mathbf{x}_i\,, y_i)\}_{1 \leq i \leq N}$ correspondant à des tirages identiquement distribués suivant P
  - Supervision : $\{y_i\}_{1 \leq i \leq N}$
  - Sauf cas particuliers (par ex. séries temporelles) on considère les données de $\mathcal{D}_N$ issues de tirages indépendants

Modélisation à partir de données : un cadre plus précis (2)

- Observations (données) avec information de supervision : $\mathcal{D}_N = \{(\mathbf{x}_i, y_i)\}_{1 \leq i \leq N}$ correspondant à des tirages identiquement distribués suivant P
  - Supervision : $\{y_i\}_{1 \leq i \leq N}$
  - Sauf cas particuliers (par ex. séries temporelles) on considère les données de $\mathcal{D}_N$ issues de tirages indépendants

→ Objectif : trouver, dans une famille $\mathcal{F}$, une fonction $f : \mathcal{X} \to \mathcal{Y}$ qui prédit y à partir de $\mathbf{x}$ et présente le risque espéré $R(f) = E_P[L(X, Y, f)]$ le plus faible
  - $L()$ est la fonction de perte (ou d'erreur)
  - E_P est l'espérance par rapport à la distribution inconnue P

- Le choix d'une fonction de perte dépend de
  - La nature du problème de modélisation : classement, régression, prédiction structurée
  - Le choix de la famille $\mathcal{F}$ et de la procédure d'optimisation associée

Fonctions de perte pour problèmes de classement

- Perte 0-1 : $L_{01}(\mathbf{x}, y, f) = \mathbf{1}_{f(\mathbf{x}) \neq y}$
  - $f(\mathbf{x}), y \in \mathcal{Y}$ ensemble fini
  - Perte nulle si prédiction correcte, perte unitaire si prédiction incorrecte
  - Si $f(\mathbf{x}) \in \mathbb{R}$ alors $L_{01}(\mathbf{x}, y, f) = \mathbf{1}_{H(f(\mathbf{x})) \neq y}$, avec $H()$ fonction échelon adéquate

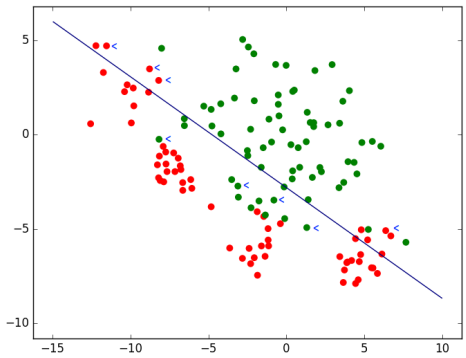


FIG. – Les flèches bleues indiquent quelques données mal classées par le modèle (frontière de discrimination linéaire, dans ce cas)

Fonctions de perte pour problèmes de classement (2)

- *Hinge loss* pour la discrimination entre 2 classes en maximisant la marge (voir chapitre SVM) : $L_h(\mathbf{x}, y, f) = \max\{0, 1 - yf(\mathbf{x})\}$ (pour $f(\mathbf{x}) \in \mathbb{R}$)
  - L_h n'est pas différentiable par rapport à f mais admet un sous-gradient
  - Des extensions existent pour le cas multi-classe et la prédiction structurée

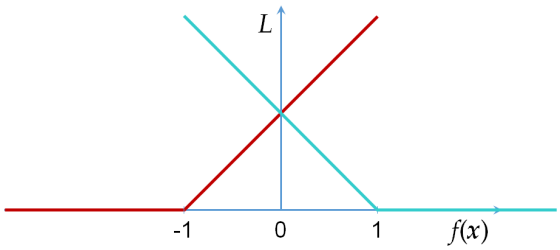


FIG. – *Hinge loss* pour $y = -1$ (en rouge) et $y = 1$ (en bleu)

Fonctions de perte pour problèmes de régression

- Perte quadratique : $L_q(\mathbf{x}, y, f) = [f(\mathbf{x}) - y]^2$
  - $f(\mathbf{x})$ est la prédiction du modèle f pour l'entrée $\mathbf{x}$
  - y est l'information de supervision (prédiction désirée) pour l'entrée $\mathbf{x}$
  - Différentiable par rapport à $f(\mathbf{x}) \Rightarrow$ une optimisation basée sur le gradient peut être appliquée directement

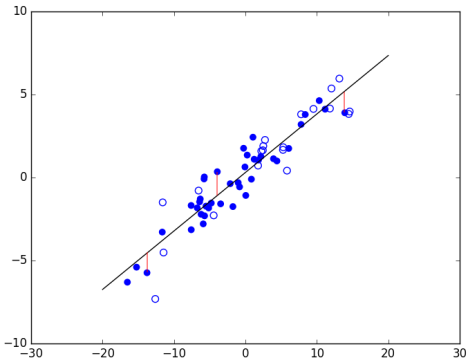
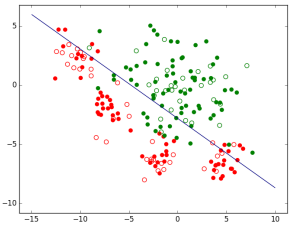


FIG. – Les traits rouges représentent des écarts entre trois prédictions d'un modèle (linéaire, dans ce cas) et les prédictions désirées correspondantes

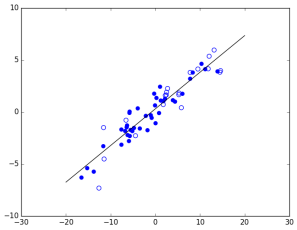
Familles paramétriques

- Modèles linéaires : prédiction = combinaison linéaire des variables explicatives
  - Exemples :

Classement :
$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$
$H(f(\mathbf{x})) \in \{-1, 1\}$



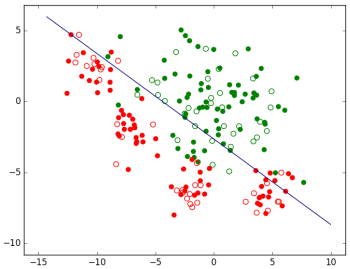
Régression :
$f(x) = w_1 x + w_0$



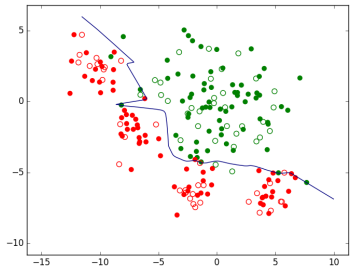
  - Peuvent s'avérer insuffisants (voir ci-dessus l'ex. de classes non linéairement séparables)
  - Utile de commencer par un modèle linéaire, ne serait-ce que pour pouvoir comparer
- Modèles polynomiaux de degré borné : la capacité d'approximation (d'une frontière pour le classement, d'une dépendance pour la régression) augmente avec le degré
- Diverses familles de modèles non linéaires, par ex. perceptrons multicouches (PMC) d'architecture donnée, etc.

Comment choisir la famille paramétrique ?

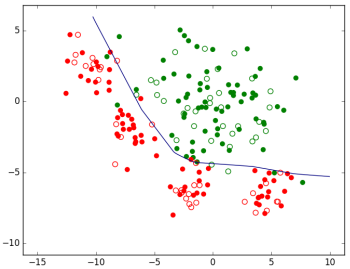
- Modèles linéaires souvent insuffisants $\rightarrow$ pourquoi ne pas choisir systématiquement une famille de capacité d'approximation aussi grande que possible ?



Err. app. 12%



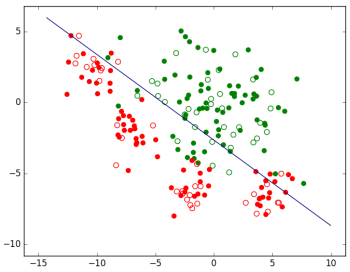
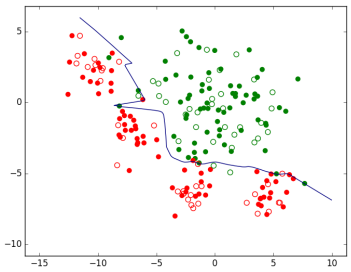
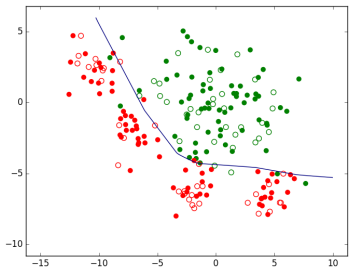
2,3%



4,5%

Comment choisir la famille paramétrique ?

- Modèles linéaires souvent insuffisants → pourquoi ne pas choisir systématiquement une famille de capacité d'approximation aussi grande que possible ?

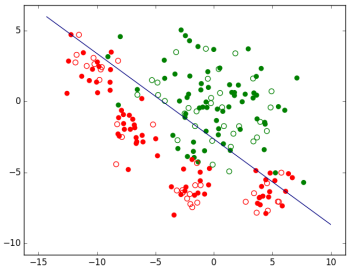
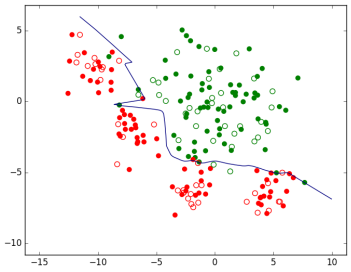
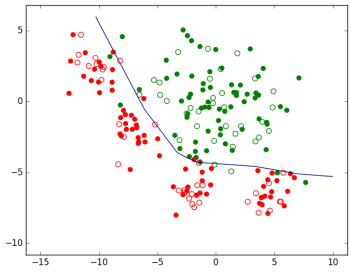




Err. app. 12%	2,3%	4,5%
Err. test 14%	6%	4,6%

Modélisation à partir de données : un cadre plus précis | Choix des familles paramétriques | 28 / 56

Comment choisir la famille paramétrique ?

- Modèles linéaires souvent insuffisants → pourquoi ne pas choisir systématiquement une famille de capacité d'approximation aussi grande que possible ?

Err. app. 12%	2,3%	4,5%
Err. test 14%	6%	4,6%

→ Risque de sur-apprentissage (*overfitting*) : erreur d'apprentissage très faible mais erreur de test comparativement élevée

⇒ Ce n'est pas avec la capacité la plus grande qu'on obtient la meilleure généralisation

→ Quel lien entre capacité et généralisation ?

Comment estimer le modèle ?

- Rappel de l'objectif : trouver, dans une famille $\mathcal{F}$ choisie, une fonction (un modèle) $f : \mathcal{X} \to \mathcal{Y}$ qui prédit y à partir de $\mathbf{x}$ et présente le risque espéré (ou théorique) $R(f) = E_P[L(X, Y, f)]$ le plus faible
- $R(f)$ ne peut pas être évalué car P est inconnue, mais on peut mesurer le risque empirique $R_{\mathcal{D}_N}(f) = \frac{1}{N} \sum_{i=1}^{N} L(\mathbf{x}_i, y_i, f)$
- Si $R(f)$ est inaccessible, comment répondre à l'objectif ?
  1. Minimisation du risque empirique (MRE) : considérer le modèle qui minimise l'erreur d'apprentissage, $f^*_{\mathcal{D}_N} = \arg\min_{f \in \mathcal{F}} R_{\mathcal{D}_N}(f)$
  2. Minimisation du risque empirique régularisé (MRER) : $f^*_{\mathcal{D}_N} = \arg\min_{f \in \mathcal{F}} [R_{\mathcal{D}_N}(f) + \alpha G(f)]$
  3. Minimisation du risque structurel (MRS) : séquence de familles de capacité qui augmente, estimation MRE dans chaque famille, choix tenant compte à la fois de $\mathcal{D}_N$ et de la capacité

Analyse des composantes du risque espéré

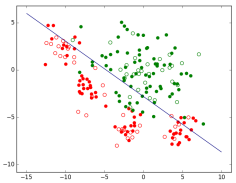
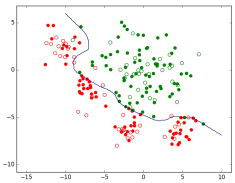
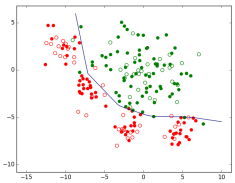
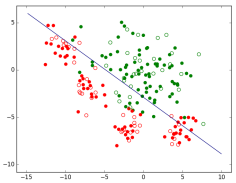
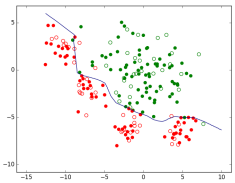
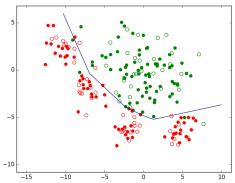
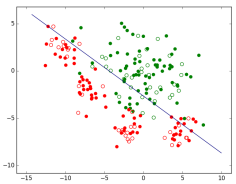
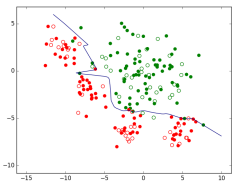
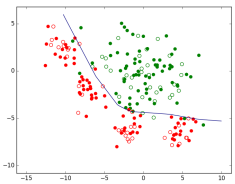
- Considérons
  - $f^*_{\mathcal{D}_N}$ la fonction de $\mathcal{F}$ qui minimise le risque empirique $R_{\mathcal{D}_N}$
  - f^* la fonction de $\mathcal{F}$ qui minimise le risque espéré R, alors

$$R(f^*_{\mathcal{D}_N}) = R^* + [R(f^*) - R^*] + [R(f^*_{\mathcal{D}_N}) - R(f^*)]$$

1. R^* est le risque résiduel (ou risque de Bayes), borne inférieure
   - Strictement positif en présence de bruit : suivant le bruit, à un même $\mathbf{x}$ peuvent correspondre plusieurs valeurs de y
2. $[R(f^*) - R^*]$ est l'erreur d'approximation (≥ 0) car $\mathcal{F}$ ne contient pas nécessairement la « vraie » dépendance
   - Nulle seulement si R^* peut être atteint par une fonction de $\mathcal{F}$
3. $[R(f^*_{\mathcal{D}_N}) - R(f^*)]$ est l'erreur d'estimation (≥ 0)
   - La fonction de $\mathcal{F}$ qui minimise le risque empirique n'est pas nécessairement celle qui minimise le risque espéré

Capacité, erreur d'approximation et erreur d'estimation

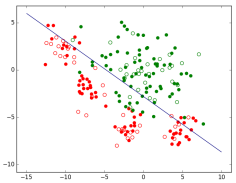
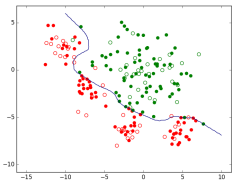
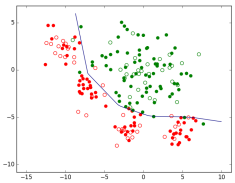
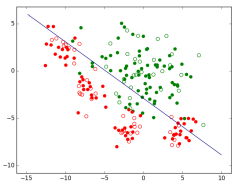
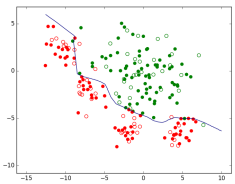
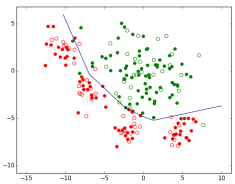
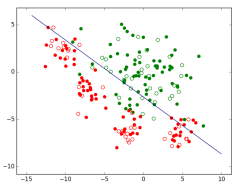
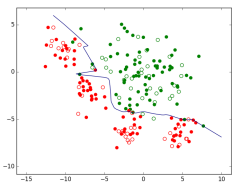
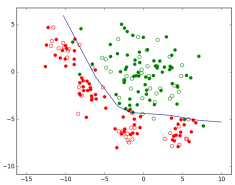
- Résultats obtenus à partir de 3 familles sur 3 échantillons différents de $\mathcal{D}_N$:

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Échantillon 1			
Échantillon 2			
Échantillon 3			

Modélisation à partir de données : un cadre plus précis | Estimation du modèle | 31 / 56

Capacité, erreur d'approximation et erreur d'estimation

- Résultats obtenus à partir de 3 familles sur 3 échantillons différents de $\mathcal{D}_N$:

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Échantillon 1			
Échantillon 2			
Échantillon 3			
Err. moyenne app.	14,4%	1,5%	4%
Err. moyenne test	9,5%	7%	5,5%
Écart-type test	0,038	0,026	0,017

Capacité, erreur d'approximation et erreur d'estimation (2)

Capacité famille linéaire (AFD) $<$ capacité PMC $\alpha = 1 <$ capacité PMC $\alpha = 10^{-5}$

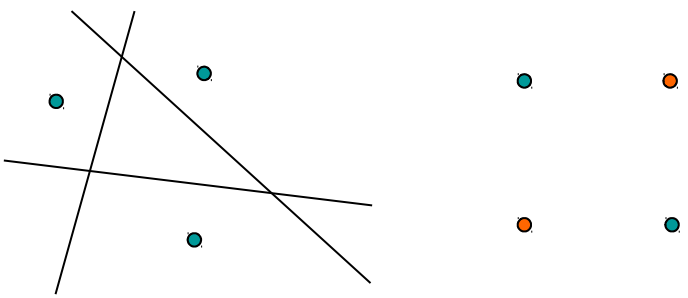
1. Famille linéaire (modèles obtenus ici par AFD)
   - Erreur d'apprentissage élevée donc capacité insuffisante pour ce problème

   $\Rightarrow$ Erreur d'approximation élevée (fort biais)
2. Famille définie par PMC 1 couche cachée de 100 neurones, avec coefficient « d'oubli » $\alpha = 10^{-5}$
   - Erreur d'approximation probablement faible car erreur d'apprentissage faible $\Rightarrow$ capacité suffisante
   - Erreur de test bien plus élevée, variance supérieure à PMC $\alpha = 1$

   $\Rightarrow$ Erreur d'estimation élevée
3. Famille définie par PMC 1 couche cachée de 100 neurones, avec coefficient « d'oubli » $\alpha = 1$
   - Somme assez faible entre erreur d'approximation et erreur d'estimation, meilleure généralisation que les deux autres familles
   - Erreur de test assez faible et proche de l'erreur d'apprentissage

Comment mesurer la capacité ?

- Considérons un ensemble de N vecteurs $\{\mathbf{x}_i\}_{1 \leq i \leq N} \in \mathbb{R}^p$ $\rightarrow$ il y a 2^N façons différentes de le séparer en 2 parties
- **Définition** : la famille $\mathcal{F}$ de fonctions $f : \mathbb{R}^p \rightarrow \{-1, 1\}$ pulvérise $\{\mathbf{x}_i\}_{1 \leq i \leq N}$ si toutes les 2^N séparations peuvent être construites avec des fonctions de $\mathcal{F}$
- **Définition** (Vapnik-Chervonenkis) : l'ensemble $\mathcal{F}$ est de VC-dimension h s'il pulvérise au moins un ensemble de h vecteurs et aucun ensemble de $h+1$ vecteurs
- Exemple : la VC-dimension de l'ensemble des hyperplans de $\mathbb{R}^p$ est $h = p + 1$
  - Dans $\mathbb{R}^2$, l'ensemble des droites pulvérise le triplet de points à gauche mais aucun quadruplet (par ex., aucune droite ne peut séparer les points bleus des rouges)



Lien entre capacité et généralisation

- La VC-dimension est une mesure intéressante de la capacité car elle permet d'obtenir une borne pour l'écart entre risque théorique et risque empirique
- **Théorème** [1] : soit $R_{\mathcal{D}_N}(f)$ le risque empirique défini par la fonction de perte $L_{01}(\mathbf{x}, y, f) = \mathbf{1}_{f(\mathbf{x}) \neq y}$; si la VC-dimension de $\mathcal{F}$ est $h < \infty$ alors pour toute $f \in \mathcal{F}$, avec une probabilité au mois égale à $1 - \delta$ $(0 < \delta < 1)$, on a

$$R(f) \leq R_{\mathcal{D}_N}(f) + \underbrace{\sqrt{\frac{h\left(\log \frac{2N}{h} + 1\right) - \log \frac{\delta}{4}}{N}}}_{B(N,\mathcal{F})} \quad \text{pour} \quad N > h$$

  - $B(N, \mathcal{F})$ diminue quand $N \uparrow$, quand $h \downarrow$ et quand $\delta \uparrow$
  - $B(N, \mathcal{F})$ ne fait pas intervenir le nombre de variables
  - $B(N, \mathcal{F})$ ne fait pas intervenir la loi conjointe P

  → résultat dans le pire des cas, intéressant d'un point de vue théorique bien que peu utile en pratique

Lien entre capacité et généralisation (2)

- Conséquences de l'existence d'une borne

$$R(f) \leq R_{\mathcal{D}_N}(f) + B(N, \mathcal{F})$$

  et de la forme de $B(N, \mathcal{F})$:
  - Famille $\mathcal{F}$ de capacité trop faible (par ex. ici modèles linéaires)
    - $\Rightarrow$ $B(N, \mathcal{F})$ faible mais $R_{\mathcal{D}_N}(f)$ (erreur d'apprentissage) élevé(e)
    - $\Rightarrow$ absence de garantie intéressante pour $R(f)$
  - Famille $\mathcal{F}$ de capacité trop élevée (par ex. ici PMC $\alpha = 10^-5$)
    - $\Rightarrow$ $R_{\mathcal{D}_N}(f)$ probablement faible mais $B(N, \mathcal{F})$ élevée
    - $\Rightarrow$ absence de garantie intéressante pour $R(f)$
  - Famille $\mathcal{F}$ de capacité « adéquate » (par ex. ici PMC $\alpha = 1$)
    - $\Rightarrow$ $R_{\mathcal{D}_N}(f)$ probablement faible et $B(N, \mathcal{F})$ plutôt faible
    - $\Rightarrow$ garantie intéressante pour $R(f)$!

Minimisation du risque empirique régularisé (MRER)

- La minimisation du risque empirique ne suffit pas à assurer une bonne généralisation, il faut maîtriser la capacité de $\mathcal{F}$ (ou la complexité du modèle)
- La régularisation est une des solutions : le modèle est obtenu en minimisant la somme entre le risque empirique $R_{\mathcal{D}_N}(f)$ et un terme $G(f)$ qui pénalise (indirectement) la capacité

$$f^*_{\mathcal{D}_N} = \arg\min_{f\in\mathcal{F}}[R_{\mathcal{D}_N}(f) + \alpha G(f)]$$

  α : hyperparamètre qui pondère le terme de régularisation

- Différentes formes pour $G(f)$, en rapport aussi avec le choix de la famille $\mathcal{F}$, par ex. :
  - $G(f) = \|\mathbf{w}\|_2^2$, $\mathbf{w}$ étant le vecteur de paramètres du modèle ; par ex. pour PMC terme « d'oubli » (*weight decay*)
  - Implicite : par ex., toujours pour PMC, terme $G(f)$ absent mais arrêt précoce (*early stopping*) de l'algorithme d'optimisation non linéaire

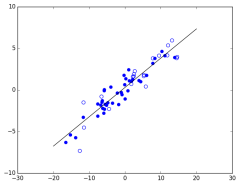
Minimisation du risque structurel (MRS)

- Une solution de maîtrise explicite de la capacité de la famille de modèles est la minimisation du risque structurel [1]

1. Définition d'une séquence $\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3 \ldots$ de familles de capacités croissantes, c'est à dire pour lesquelles $h_1 < h_2 < h_3 \ldots$
2. Pour $i \in \{1, 2, 3 \ldots\}$, minimisation dans chaque famille du risque empirique $f^{(i)*}_{\mathcal{D}_N} = \arg\min_{f\in\mathcal{F}_i} R_{\mathcal{D}_N}(f)$
3. Tenant compte de la borne trouvée pour le risque espéré R, sélection de $f^{(i)*}_{\mathcal{D}_N}$, $i \in \{1, 2, 3 \ldots\}$, qui minimise $R_{\mathcal{D}_N}(f^{(i)*}_{\mathcal{D}_N}) + B(N, \mathcal{F}_i)$

Comment minimiser le risque empirique (régularisé) ?

- Dans une famille paramétrique $\mathcal{F}$, un modèle est défini par les valeurs d'un ensemble de paramètres, par ex.
  - Modèle linéaire pour la régression $y = ax + b$: a et b



  - Perceptron multi-couches d'architecture donnée : poids des connexions de la (des) couche(s) cachée(s) et de la couche de sortie

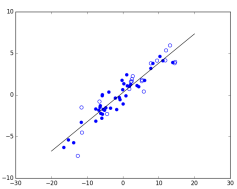
→ Optimisation pour trouver les valeurs qui minimisent le critère (MRE, MRER)
- Solution analytique directe : cas assez rare, par ex. certains modèles linéaires
- Algorithmes itératifs, par ex.
  - Optimisation quadratique sous contraintes d'inégalité : SVM
  - Optimisation non linéaire plus générale : PMC, réseaux profonds

Exemple : régression linéaire

- Problème de régression avec $\mathcal{X} = \mathbb{R}^p$, $\mathcal{Y} = \mathbb{R}$, $\mathcal{D}_N = \{(\mathbf{x}_i, y_i)\}_{1 \leq i \leq N}$
- Famille de modèles linéaires $\hat{y} = w_0 + \sum_{j=1}^{p} w_j x_{ji}$, où $\hat{y}$ est la prédiction du modèle
- Sous forme matricielle : $\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$, où $\mathbf{X}$ est la matrice $N \times (p+1)$ dont les lignes sont les observations de $\mathcal{D}_N$ et les colonnes correspondent aux variables (sauf pour la dernière qui est une colonne de 1 et permet d'inclure w_0 dans $\mathbf{w}$)
- On cherche le modèle (défini par le vecteur de paramètres $\mathbf{w}^*$) qui minimise
  - MRE : l'erreur quadratique totale $\sum_{i=1}^{N} (\hat{y}_i - y_i)^2$ sur $\mathcal{D}_N$

    → Solution $\mathbf{w}^* = \mathbf{X}^+ \mathbf{y}$, où $\mathbf{X}^+$ est la pseudo-inverse Moore-Penrose de $\mathbf{X}$

    - Si $\mathbf{X}^T\mathbf{X}$ est inversible, alors $\mathbf{X}^+ = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$



  - MRER : la somme entre l'erreur quadratique sur $\mathcal{D}_N$ et un terme de régularisation, par ex. (cas particulier de régularisation Tikhonov), $\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 + \|\mathbf{w}\|_2^2$

    → Solution $\mathbf{w}^* = (\mathbf{X}^T\mathbf{X} + \mathbf{I}_{p+1})^{-1}\mathbf{X}^T\mathbf{y}$, où $\mathbf{I}_{p+1}$ est la matrice unité de rang $p+1$

Modélisation décisionnelle : que faut-il retenir ?

- Construire un modèle décisionnel à partir de données : supervision nécessaire
- Objectif : obtenir le modèle qui présente la meilleure généralisation
- Estimer la généralisation : non à partir de l'erreur d'apprentissage
- Chercher le bon compromis entre minimisation de la capacité de la famille de modèles et minimisation de l'erreur d'apprentissage

Plan du cours

Comment estimer le risque espéré

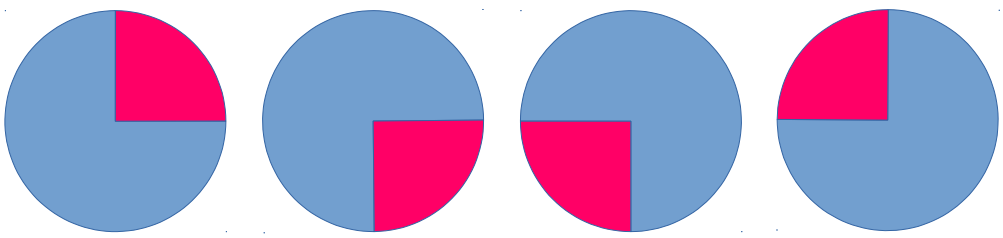
1. A partir du risque empirique et en tenant compte de bornes de généralisation : $R(f^*_{\mathcal{D}_N}) \leq R_{\mathcal{D}_N}(f^*_{\mathcal{D}_N}) + B(N, \mathcal{F})$
   → Lorsqu'elle existe, la borne est en général trop élevée pour être utile en pratique
2. Par l'erreur sur des données de test, non utilisées pour l'apprentissage
   - Les observations disponibles avec information de supervision sont partitionnées (par échantillonnage uniforme, en général) en données d'apprentissage (70-80%) et données de test (20-30%)
     - Apprentissage (estimation) du modèle sur les données d'apprentissage
     - Estimation du risque espéré par l'erreur de ce modèle sur les données de test

   → Difficultés de cette approche :
     - La mise de côté des données de test réduit le nombre de données utilisées pour l'apprentissage
     - Cet estimateur du risque espéré a une variance élevée (un autre partitionnement produira d'autres ensembles d'apprentissage et de test)

   → Validation croisée (*cross-validation*) : plusieurs partitionnements apprentissage | test, obtenir à chaque fois un modèle sur les données d'apprentissage et l'évaluer sur les données de test associées, employer la moyenne comme estimation du risque espéré
   ⇒ estimateur de variance plus faible,
   . . . tout en utilisant mieux les données disponibles !

Validation croisée

1. Méthodes exhaustives :
   - *Leave p out* (LPO) : $N - p$ données pour l'apprentissage et p pour la validation ⇒ C^p_N découpages possibles donc C^p_N modèles à apprendre ⇒ coût excessif
   - *Leave one out* (LOO) : $N - 1$ données pour l'apprentissage et 1 pour la validation ⇒ $C^1_N = N$ découpages possibles (donc N modèles) ⇒ coût élevé
2. Méthodes non exhaustives :
   - *k-fold* : partitionnement fixé des N données en k parties, apprentissage sur $k - 1$ parties et validation sur la k-ême ⇒ k modèles seulement (souvent $k = 5$ ou $k = 10$)



   - Échantillonnage répété (*shuffle and split*) : échantillon aléatoire de p données pour le test (les autres $N - p$ pour l'apprentissage), on répète cela k fois ⇒ k modèles

Validation croisée : quelle méthode préférer ?

- LPO très rarement employée car excessivement coûteuse
- LOO *vs k-fold* : *k-fold* préférée en général
  - LOO plus coûteuse car $N \gg k$
  - Variance en général supérieure pour LOO
  - Estimation *k-fold* pessimiste car chaque modèle apprend sur $\frac{k-1}{k} N < N - 1$ données
- *Shuffle and split vs k-fold*
  - Pour *k-fold* le nombre de modèles (k) est lié à la proportion de données de test ($1/k$), *shuffle and split* moins contraignante
  - Pour *shuffle and split* certaines données ne sont dans aucun échantillon alors que d'autres sont dans plusieurs échantillons
- Quelle que soit la méthode, tous les partitionnements peuvent être explorés en parallèle (sur processeurs multi-cœur ou plateformes distribuées)

Validation croisée : précautions à prendre

- Problème de classement avec classes (très) déséquilibrées : pour s'assurer de conserver les rapports entre les classes dans tous les découpages, utiliser
  - Un partitionnement adapté pour *k-fold* (par ex. `StratifiedKFold` dans Scikit-learn)
  - Un échantillonnage stratifié pour *shuffle and split* (par ex. `StratifiedShuffleSplit` dans Scikit-learn)
  - LOO peut être employée telle quelle
- Observations qui ne sont pas indépendantes
  - Séries temporelles : les observations successives sont corrélées, le découpage doit être fait par séquences sur les observations ordonnées et non après *shuffle* sur les observations individuelles
  - Données groupées : dans un même groupe, les observations ne sont pas indépendantes ; les données de test doivent provenir de groupes différents de ceux dont sont issues les données d'apprentissage

Évaluation pour problèmes de classement à coûts asymétriques

- Estimation du risque espéré d'un modèle de classement : taux de mauvais classement sur les données de test
  - Taux de mauvais classement ← fonction de perte L_{01}
  - → coût symétrique : même coût si le modèle se trompe dans un sens ou dans l'autre
- De nombreux problèmes présentent des coûts asymétriques, par ex.
  - Pour un cargo, la non détection d'un autre navire par le radar peut mener à une collision, alors qu'une fausse alerte provoque seulement un ralentissement temporaire
  - La non détection de la maladie grave d'un patient est dramatique, alors que la détection erronée d'une telle maladie pour un patient sain est moins problématique

⇒ Comment examiner les caractéristiques de différents modèles lorsque les coûts sont asymétriques, sans fixer le « degré » d'asymétrie ?

Terminologie pour la discrimination entre 2 classes

- Une classe peut être considérée la classe « d'intérêt »
- Le modèle appris est vu comme le « détecteur » de la classe d'intérêt
- Pour un tel détecteur appris, les cas suivants peuvent être constatés :

	Classe présente	**Classe absente**
Classe détectée	Vrai Positif	Faux Positif
Classe non détectée	Faux Négatif	Vrai Négatif

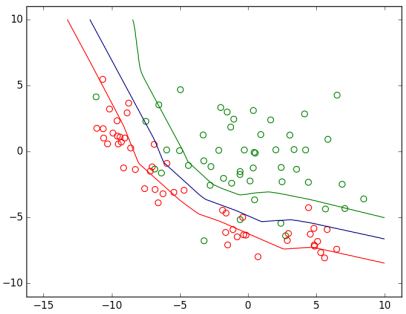
- On définit les mesures suivantes :

  Taux de vrais positifs (ou sensibilité) $= \frac{\text{Vrais Positifs}}{\text{Total Positifs}} = \frac{\text{VP}}{\text{VP + FN}}$

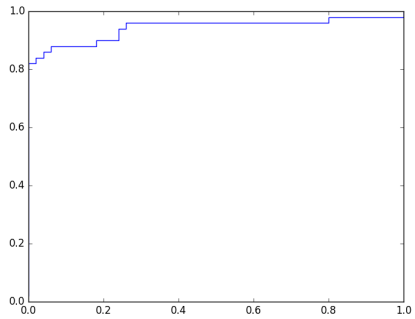
  Taux de faux positifs (ou $1-$ spécificité) $= \frac{\text{Faux Positifs}}{\text{Total Négatifs}} = \frac{\text{FP}}{\text{VN + FP}} = 1 - \frac{\text{VN}}{\text{VN + FP}}$
- Idéalement
  1. Toutes les détections positives devraient correspondre à de vrais positifs : pas de faux négatifs (FN $= 0$), ou taux de vrais positifs $= 1$
  2. Ce qui n'est pas détecté devrait correspondre aux seuls vrais négatifs : pas de faux positifs (FP $= 0$), ou taux de faux positifs $= 0$

Courbes ROC pour discrimination entre 2 classes

- Modèle : en général décrit par un vecteur de paramètres $\mathbf{w}$ (par ex. poids connexions pour PMC) et un seuil b (par ex. sur la probabilité de la classe d'intérêt)
- Courbe ROC : taux de vrais positifs (en ordonnée) fonction du taux de faux positifs (en abscisse), la variable étant le seuil
- Pour un $\mathbf{w}$ fixé, peut-on réduire en même temps FN et FP en faisant varier le seuil ?



Frontières pour 3 valeurs du seuil de détection

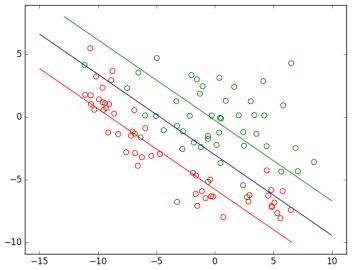
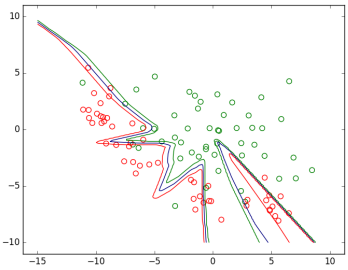
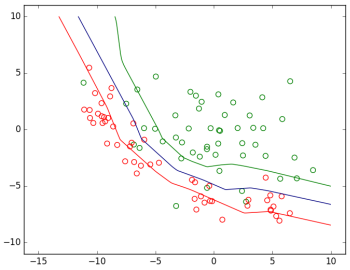


Courbe ROC associée

$\Rightarrow$ si on augmente le taux de vrais positifs, le taux de faux positifs augmente également !

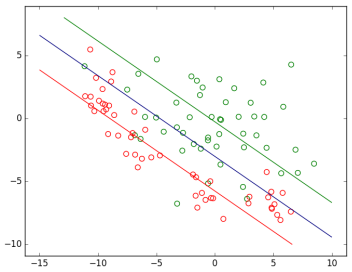
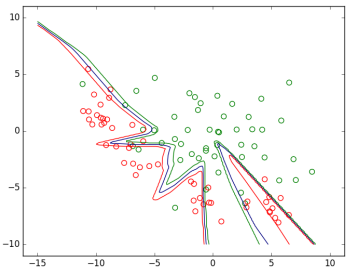
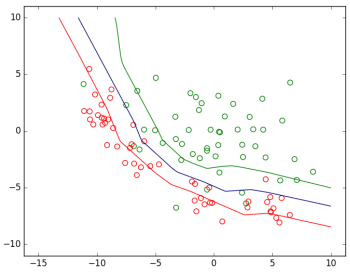
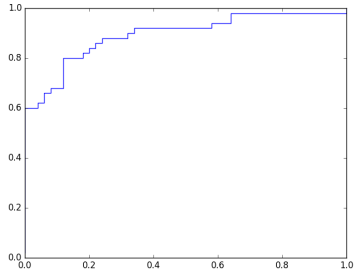
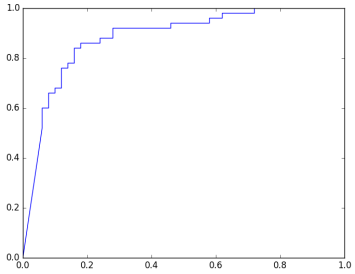
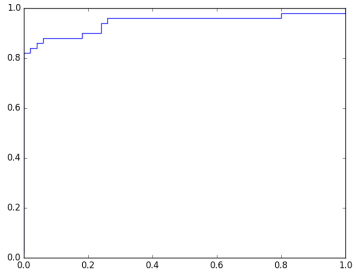
Comparaison de modèles à travers les courbes ROC

- Comparaison globale par rapport au domaine de variation du seuil :

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Discrimination			

Comparaison de modèles à travers les courbes ROC

- Comparaison globale par rapport au domaine de variation du seuil :

	AFD	PMC $\alpha = 10^{-5}$	PMC $\alpha = 1$
Discrimination			
Courbes ROC			
AUC	0,89	0,88	0,94

Comparaison de modèles à travers les courbes ROC (2)

- Un outil de comparaison globale est l'aire sous la courbe ROC (*area under curve*, AUC) : plus l'aire sous la courbe ROC est élevée, meilleur est le modèle
- Si valeurs AUC proches ou pour objectifs plus précis : comparaison des taux de vrais positifs (sensibilité) à taux de faux positifs (spécificité) donné(e)s

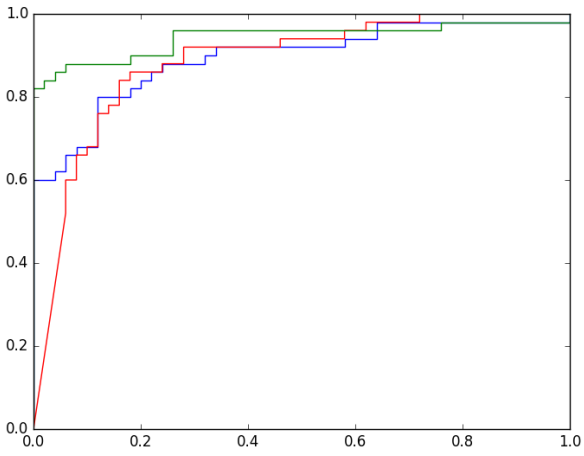


FIG. – Courbes ROC : AFD en bleu, PMC $\alpha = 10^{-5}$ en rouge, PMC $\alpha = 1$ en vert

Plan du cours

Sélection de modèles

- Dans l'estimation d'un modèle, par ex. par MRER

$$f^*_{\mathcal{D}_N} = \arg\min_{f \in \mathcal{F}} [R_{\mathcal{D}_N}(f) + \alpha G(f)]$$

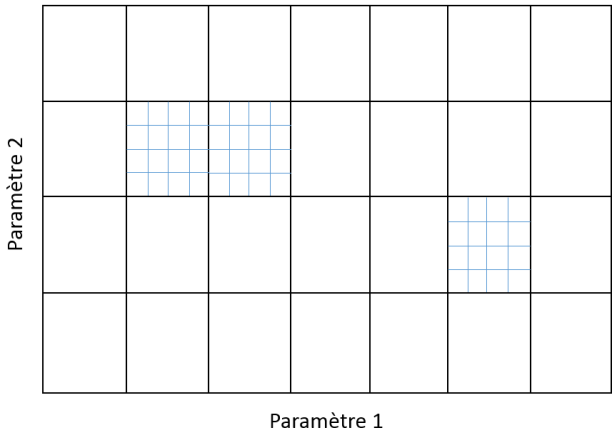
plusieurs hyperparamètres interviennent :
  - La pondération de la régularisation, α
  - Le critère de régularisation $G(f)$
  - Autres paramètres qui ont un impact direct sur $\mathcal{F}$, par ex. l'architecture pour un PMC, le type de noyau (et la variance du noyau) pour une SVM, etc.
- Comment choisir de « bonnes » valeurs pour ces hyperparamètres ?
  - → Procédures de recherche qui explorent l'espace des valeurs des paramètres, de façon systématique ou aléatoire
  - Les modèles obtenus pour différentes valeurs des hyperparamètres sont comparés à travers leurs scores de validation croisée
  - Une fois trouvé le meilleur modèle, son risque espéré est estimé sur des données de test qui n'ont servi ni à la recherche des paramètres, ni à celle des hyperparamètres !

Recherche systématique : *grid search*

- Pour trouver les meilleures valeurs des hyperparamètres, une première possibilité est d'explorer l'espace des hyperparamètres de façon systématique
- Recherche en grille (*grid search*) :
  1. Définition d'intervalles et de pas de variation pour les hyperparamètres numériques (par ex. constante de régularisation α, variance de noyau RBF)
  1. Définition d'ensembles de valeurs pour les hyperparamètres nominaux (par ex. architectures PMC, critères de régularisation, types noyaux SVM)
  2. Exploration systématique de l'espace des hyperparamètres
  3. Choix des valeurs pour lesquelles le modèle obtenu présente les meilleures performances de validation croisée
- Estimation du risque espéré (erreur de généralisation) du modèle obtenu : sur des données non encore utilisées !

Recherche systématique : *grid search* (2)

- Lorsque seuls des hyperparamètres continus sont présents, on obtient une grille = combinaisons de valeurs à tester pour les m paramètres $\Rightarrow$ grille de dimension m



- Tous les points de la grille peuvent être explorés en parallèle !
- Plusieurs niveaux de « finesse » $\rightarrow$ recherche hiérarchique : exhaustive suivant la grille grossière, puis là où les résultats sont meilleurs on affine suivant le(s) niveau(x) plus fin(s) $\rightarrow$ augmentation du rapport qualité des résultats / coût

Recherche aléatoire : *randomized parameter optimization*

- Des connaissances *a priori* permettent de privilégier certains intervalles de variation $\rightarrow$ générer des valeurs conformes à ces connaissances $\rightarrow$ meilleure efficacité qu'avec *grid search* non hiérarchique
- Le coût peut être maîtrisé en fixant le nombre d'échantillons à générer
- Modalités d'échantillonnage
  1. Hyperparamètres numériques à valeurs continues (par ex. α) : loi d'échantillonnage (par ex. loi normale d'espérance et variance données)
  2. Hyperparamètres numériques à valeurs discrètes (par ex. nombre de neurones cachés) : loi d'échantillonnage (par ex. loi uniforme sur intervalle donné)
  3. Hyperparamètres variables nominales : liste des valeurs (modalités) possibles $\rightarrow$ loi uniforme sur ces valeurs
- Échantillons générés en considérant les hyperparamètres indépendants

Évaluation et sélection de modèles : que faut-il retenir ?

- Estimation du risque espéré (erreur de généralisation) sur des données non utilisées pour l'apprentissage
- Validation croisée : meilleure estimation qu'un seul découpage apprentissage | test
- Courbes ROC : comparaison plus globale de modèles de classement
- Meilleures valeurs pour les hyperparamètres : recherche systématique ou aléatoire, comparaison des modèles par validation croisée
- Si validation croisée employée pour sélectionner le meilleur modèle, estimation du risque espéré du modèle retenu sur des données non encore utilisées

Références I


O. Bousquet, S. Boucheron, and G. Lugosi.
Introduction to Statistical Learning Theory, volume Lecture Notes in Artificial Intelligence 3176, pages 169–207.
Springer, Heidelberg, Germany, 2004.


O. Chapelle, B. Schölkopf, and A. Zien, editors.
Semi-Supervised Learning.
MIT Press, Cambridge, MA, 2006.


I. Goodfellow, Y. Bengio, and A. Courville.
Deep Learning.
MIT Press, 2016.
`http://www.deeplearningbook.org`.


B. Schölkopf and A. Smola.
Learning with Kernels.
MIT Press, 2002.

Références II


L. Yang, H. Rodriguez, M. Crucianu, and M. Ferecatu.
Fully convolutional network with superpixel parsing for fashion web image segmentation.
In *Proc. 23rd Intl. Conf. MultiMedia Modeling, Reykjavik, Iceland*, pages 139–151, 2017.