Analyse,Classification et Indexation de données

Notes

Robin Montfermé

Master 1 Informatique 2019-2020

Table des matières

Introduction

Chargé de Cours :
Jean-Phillipe Domenger- `jean-philippe.domengerarobaselabri.fr`
Modalité d'examen :

— CC : DS + TP (fin de semstre)
— Examen

Supports

Notes Partielles : masterinfo.emi.u-bordeaux.fr

Objectifs

À partir d'un ensemble de points et d'un vecteur v $\in \mathbb{R}^n$ de paramètres, il faut réussir à que les points appartiennent à une classe(x *et* $y \in$ *classe bleu*)

Contenu

Cours magistraux : présentation des concepts.
TD : implémentations des concepts. Utilisation de MATLAB.

Bibliographie

— *Pattern classification*, R.O.Duda, chez John Wiley&Sons .

Chapitre 1

Introduction à la classification de données

1.1 Système et modèles

1.1.1 Système numérique

Un système numérique est défini par 3 éléments :
— Des capteurs, recevant des données **continues**.
— Une unité de calcul, traitant des données **discrètes**.
— Un stockage.

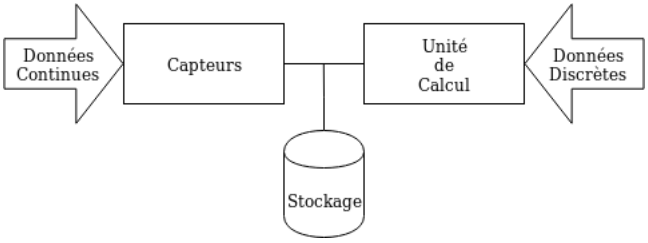


FIGURE 1.1 – Schéma d'un système numérique

1.1.2 Modèle

Comment traiter les données fournies par les capteurs au système ?
Tout algorithme est basé sur un modèle. Ce que la classification cherche à faire c'est d'intégrer à un modèle, une méthode de prise de décision, afin de créer un algorithme capable de prendre une décision.

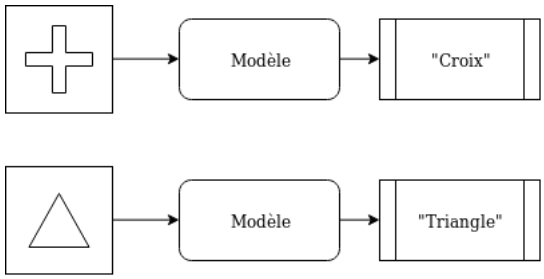


FIGURE 1.2 – Example de classification

1.2 Apprentissage

Pour qu'un modèle de classification soit créé, il faut passer par une phase d'apprentissage. Il y a 3 type d'apprentissage.

1.2.1 L'apprentissage supervisé

L'apprentissage supervisé consiste, à créer une observation. Une observation est un ensemble de données associées à des annotations.

Observation = (donnée 1, classe x),(donnée 2, classe y),(donnée 3, classe z)
Exemple d'observation

Une **annotation** est une indication qui identifie clairement à quelle classe appartiennent des données.
Les **données** sont des valeurs numériques obtenues via les capteurs. Elles permettent de créer des **descripteurs**.
Un **descripteur ou feature**[1] est un ensemble de données permettant d'identifier la classe à laquelle appartiennent les éléments observées.

1. Il s'agit des la notion de paramètres evoquée en introduction.

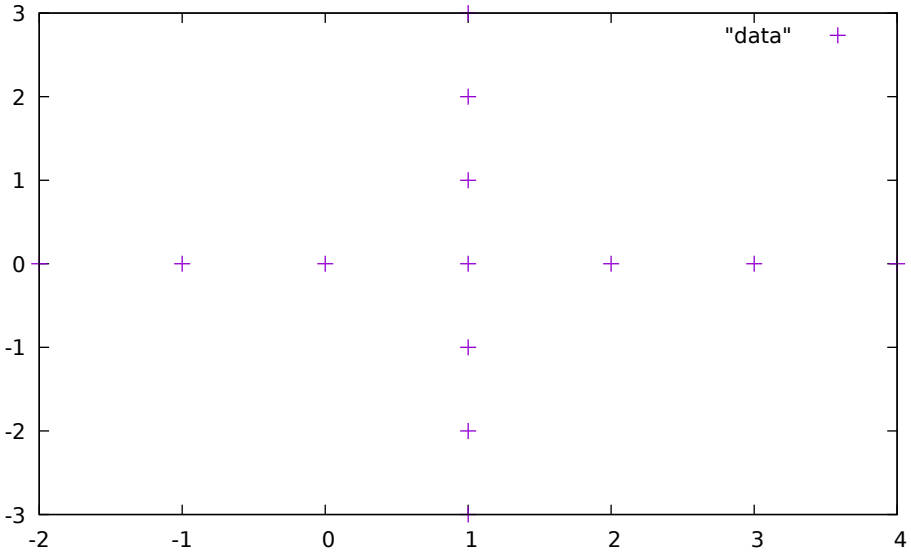


FIGURE 1.3 – Exemple de données

Les données représentées ici (1.3) peut être écrit de la façon suivante :

$$[(-1, 0), (0, 1), ...] \in \mathbb{R}^n$$

Cet ensemble $\mathbb{R}^n$ correspond à l'ensemble de descripteurs.
Ces ensembles de données annotées forment la **Vérité Terrain**.
En conclusion, l'apprentissage supervisé consiste à fournir au modèle une **Verité Terrain**.

1.2.2 L'apprentissage non-supervisé

Pour cet apprentissage on ne fourni aucune annotation au modèle.

$$\{x \in \mathbb{R}^n\}$$

Le modèle n'obtient que des données

Cependant, on sait combien de classe il doit y avoir. On défini les classes à partir des données obtenues des échantillons observés.

1.2.3 L'aprentissage semi-supervisé

Il s'agit simplement d'un mélange des précédent type d'apprentissage. La **Vérité Terrain** est ici partielle

On sait :
(x_1,c_1), (x_2,c_2)..., (x_n,c_2)..., x_{n+1},x_{n+2},...
On souhaite :
(x_1,c_1), (x_2,c_2)..., (x_n,c_2)..., $\underline{(x_{n+1},c_2)},\underline{(x_{n+2},c_1)}$,...

1.2.4 Résumé

- Les **descripteurs ou features** sont des ensemble de donnée appartenant à $\mathbb{R}^n$.
- Les **classes** sont des données discrètes qui peuvent :
  - appartenir à $\mathbb{N}$ si il s'agit de nombre.
  - appartenir à l'ensemble des chaînes de caractères.
- Les **observations** sont un ensemble de descripteurs et de classes.

1.3 L'exemple de l'usine de tri de poisson

Imaginons le système suivant :

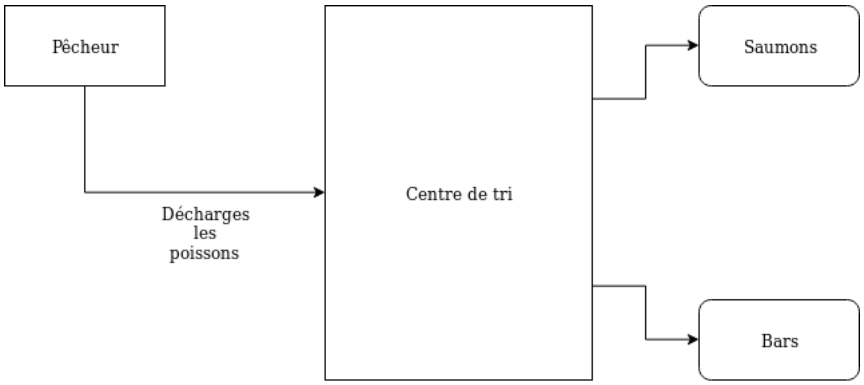


FIGURE 1.4 – Usine de tri de poisson

On veut faire en sorte que les poissons amenés par le pêcheurs soient trié dans les bacs "saumons" et "bars" [2]. Ici on a donc identifié la classe "Saumon" et la classe "Bar".ur

1.3.1 Problème de pertinence

Il y a beaucoup de descripteurs que l'on peut obtenir à partir des poissons, le poids ou l'apparence par exemple. Mais l'important est que les descripteurs soient **pertinents**.

2. Le pêcheur n'a pêcher que ces espèces de poisson.

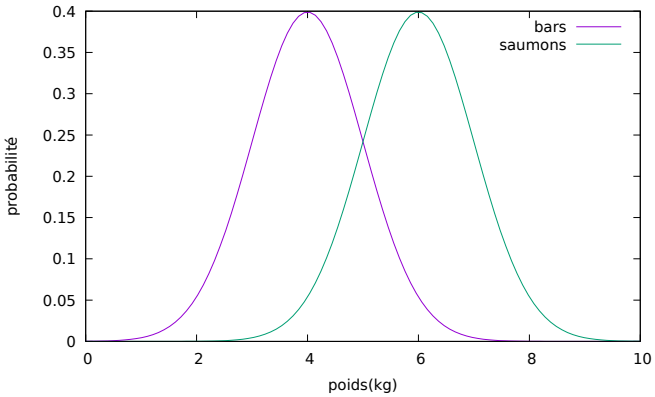


FIGURE 1.5 – Cas où le poids (seul) n'est pas un descripteur pertinent.

Sur ce graphique (1.5) on voit que les poids mesurés chez les deux espèces de poissons peuvent être très similiaires, ce qui rend l'utilisation du poids en tant que descripteur n'est pas très pertinent [3].

1.3.2 Problème de dimension

On a vu que certains paramètres ne sont pas pertinents, mais il se peut que certains descripteur posent des problèmes de dimension.
Si l'usine est équippée d'une caméra pour scanner les poissons, les données récoltés dépendent de la résolution de la caméra.

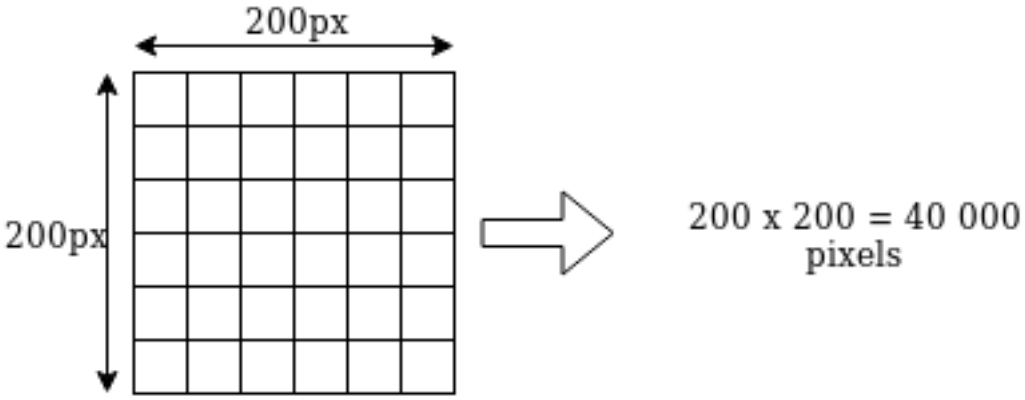


FIGURE 1.6 – Pixels de la caméra

Avec l'image obtenu de la caméra on a 200 par 200 = 40 000 pixel.s On rappel qu'un descripteur est un vecteur, on a donc un descripteur $p \in \mathbb{R}^{40000}$

3. Le problème vient du fait que les courbes se croisent trop.

Cela pose un problème de **dimension**. Les données sont trop éparpillées et il se peut que lorsque en réalité, il y ait deux poissons de même classe mais étant dans une disposition différente, que ces derniers ne soient pas rassemblé sous la même classe(voir 1.7).

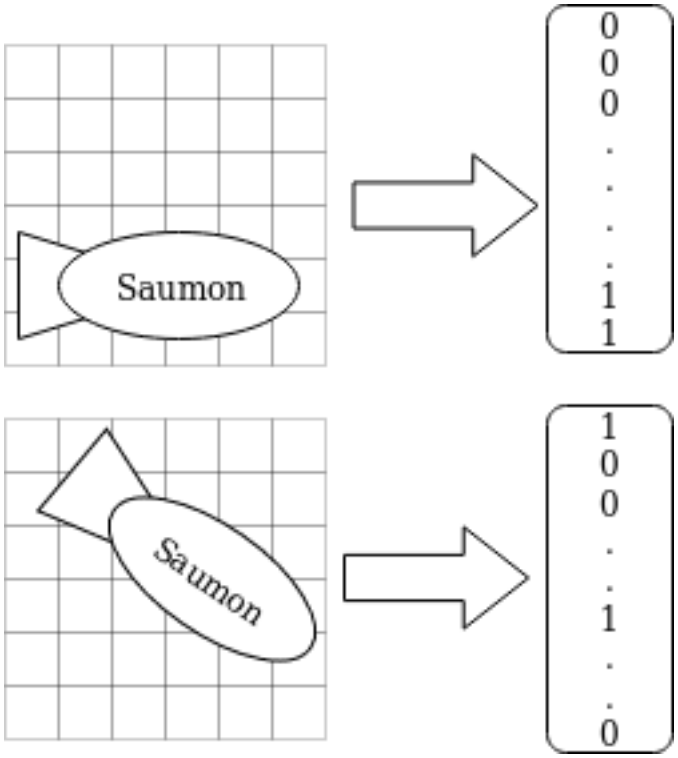


FIGURE 1.7 – Deux poissons de même classes offrant des descripteurs très différents

Il faut donc réduire la dimension de ce descripteur.

1.3.3 Réduction de la dimension des descripteur

Tout comme il y a de nombreux descripteurs, il y a de nombreux moyens de les réduire. Dans notre exemple, on va faire un traitement de l'image afin d'identifier ce qui est un poisson et ce qui ne l'est pas, puis créer la **Boîte Englobante Orientée** de poisson.

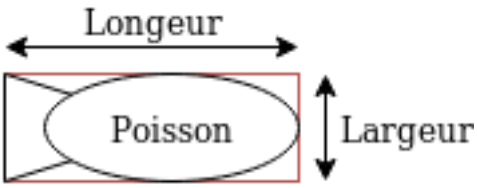


FIGURE 1.8 – Bôite englobante orienté pour les poissons

La boîte englobante orienté permet plusieurs choses :
— On obtient la longeur et la largeur du poisson, on passe de $\mathbb{R}^{40000}$ à $\mathbb{R}^2$, il n'y a plus de problème de dimension.
— Peut importe la position du poisson sur l'image on peut extraire des descripteur **pertinents**.

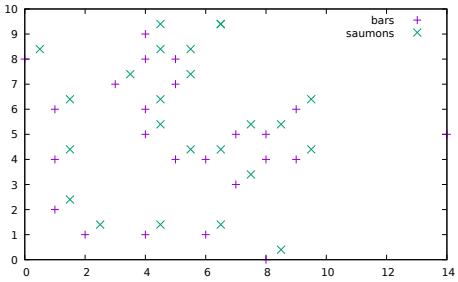
1.3.4 Sensibilité des descripteurs

Un autre point important est de savoir si les descripteurs sont sensibles ou non à certaines transformations.
Exemple : l'image du poisson
— Est sensible : au zoom (changement des dimensions mesurées)
— N'est pas sensible : à la rotation (Boîte Englobante Orientée)

1.3.5 Comment s'assurer de la qualité du descripteur ?

Verification graphique

Si on représente les données sur un graphes on va vouloir :
— que les points soient regroupe de manière dense
— que l'on puisse séparer les deux classes



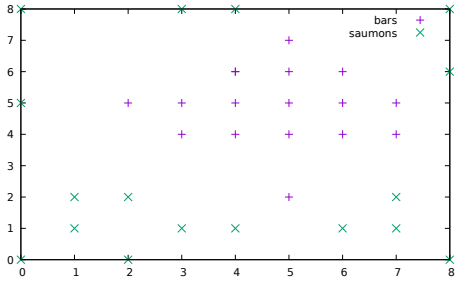
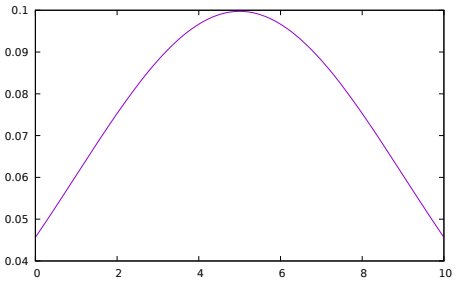


FIGURE 1.9 – Exemple de mauvais descripteurs (à gauche)

FIGURE 1.10 – Exemple de bon descripteur (à droite)

Variation de descripteur

On veut aussi que le descripteur ai peu de variation :



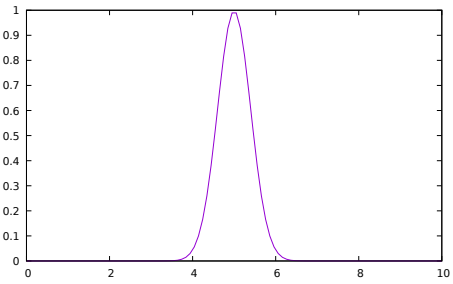


FIGURE 1.11 – Exemple de mauvaise variation (à gauche)

FIGURE 1.12 – Exemple de bonne variation (à droite)

Rappel sur la pertinence

On rappelle aussi qu'il faut que le descripteur doit être pertinent [4] :

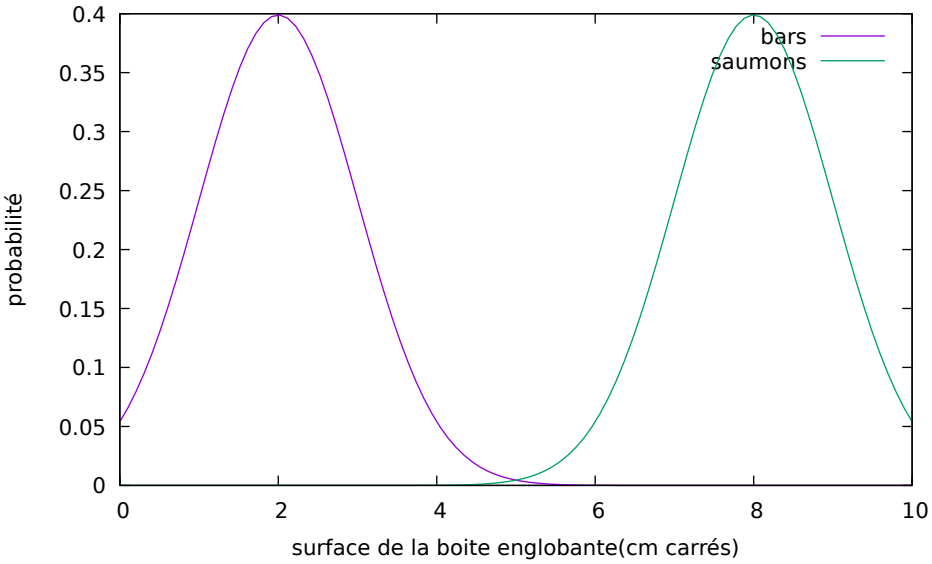


FIGURE 1.13 – La surface est plus pertinente que le poids(voir fig 1.5).

4. On remarquera au passage que la dimension de la surface de la boite englobante est $\mathbb{R}^1$

Ajout de paramètre

On peut rajouter des paramètre. Par exemple, la luminance de l'image. Cela va augmenter la dimension du descripteur. Pour bien faire il faut que ce nouveau paramètre soit **indépendant** des anciens. La luminance et la surface sont des paramètres indépendant l'un de l'autre.

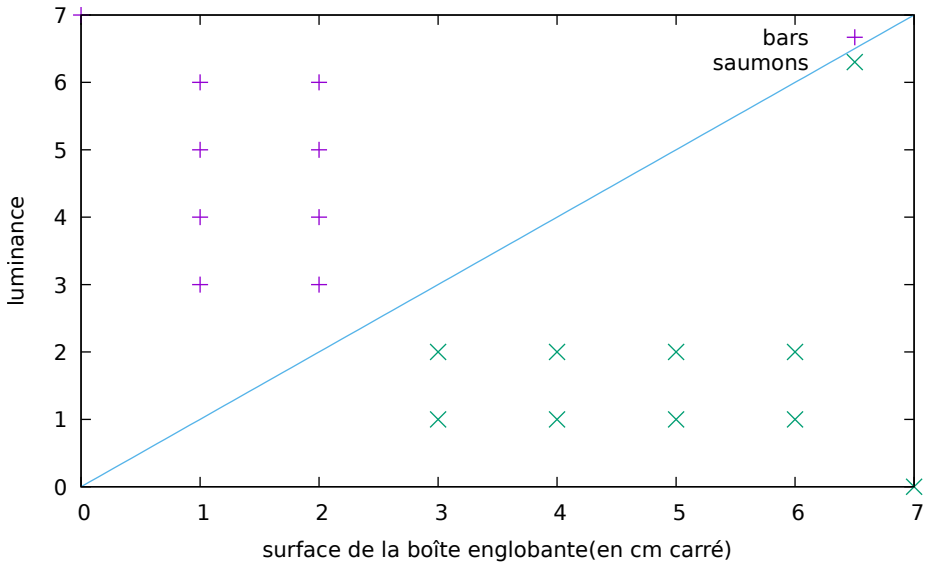


FIGURE 1.14 – L'ajout de la luminance rend le descripteur très performant

On voit bien que sur 1.14, la montée en dimension aide à la séparation.

1.3.6 Injection de données non-numériques

Il se peut que pour certaines raison il y ait besoin d'injecter des données, non quantifiable dans le système. Par exemple, si le classifieur de l'usine doit être entraîné pendant la saison des amours du Saumon, alors il faudra peut être appliquer un seuil de décision différent qu'en temps normal.

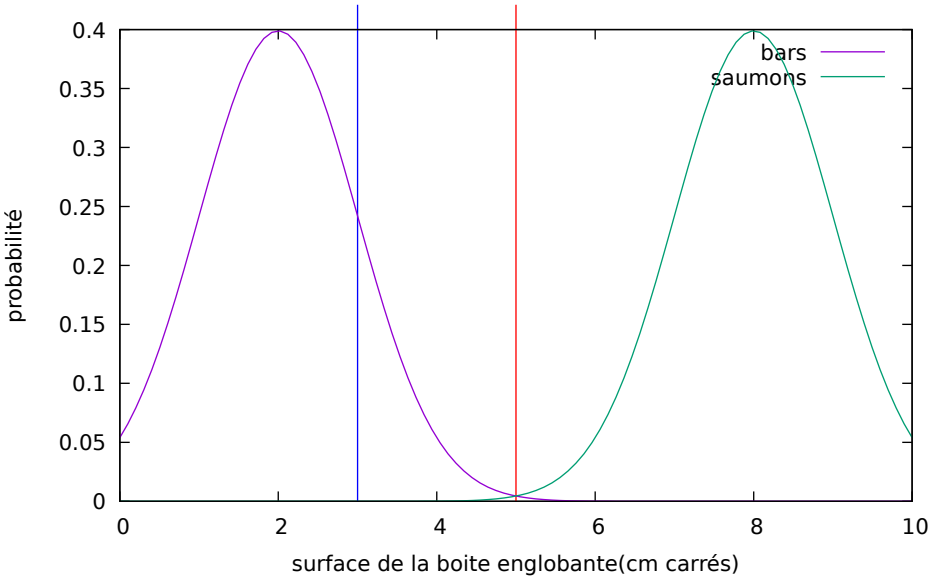


FIGURE 1.15 – En rouge seuil en temps normal, en bleu seuil pendant la saison des amours du saumon

On favorise ici un poisson plus que l'autre

1.3.7 Résumé et remarques

Résumé

— Il ne faut pas de descripteur de trop grande dimension
— Rajouter des descripteur aide à la séparation
— Les descripteurs doivent être pertinents
— Il ne doit pas avoir de corrélation entre descripteurs
— Les descripteur peuvent être mis en relation sans corrélation

Remarques

Si un classifieur obtient 0% de marge d'erreur [5] alors il est trop adapté à la vérité terrain. Ce n'est pas une bonne chose car la vérité terrain peut être

5. La marge d'erreur s'évaleur en fonction du pourcentage de la donnée A classé en tant que B.

amené à changer.
Le classifieur doit être adapté à la verité terrain.

Chapitre 2

Régression linéaire

2.1 Définition

Rappel sur l'observation

Une observation est le fait d'associer des descripteurs/features à une classe/annotation. On rappelle que $Feature \in \mathbb{R}^n$ et que $classe \in \mathbb{N} | classe \in \mathbb{R}$ Entraîner un classifieur a pour but d'associer des features à des classes afin de trouver une **fonction F** qui pourra *deviner* la classe correspondant à une feature.

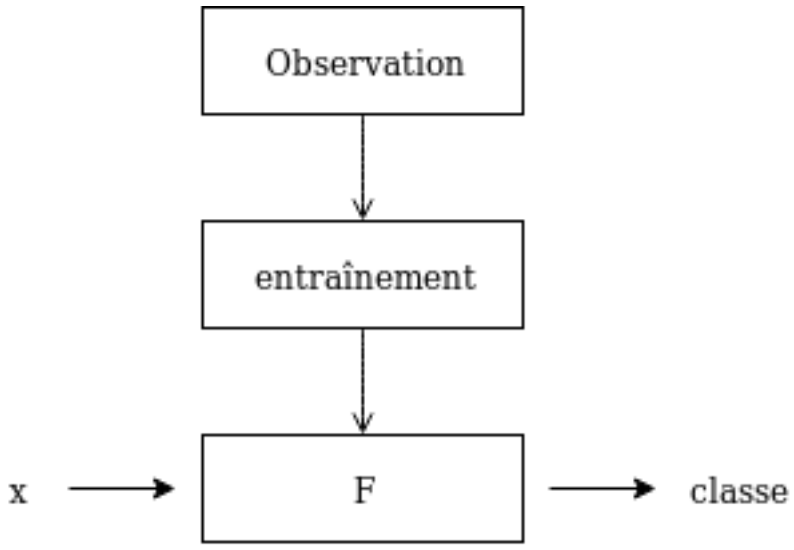


FIGURE 2.1 – L'entraînement produit une fonction F

La fonction $F : \mathbb{R}^n \to \mathbb{R}$ obtenue par entraînement du classifieur va prendre

en entrée une valeur x et va prédire une valeur y et ce à partir de données non-présentes dans l'observation [1]. **Régression : Prendre quelque chose en entrée et prédire la sortie**.

Notions liés à la Regression

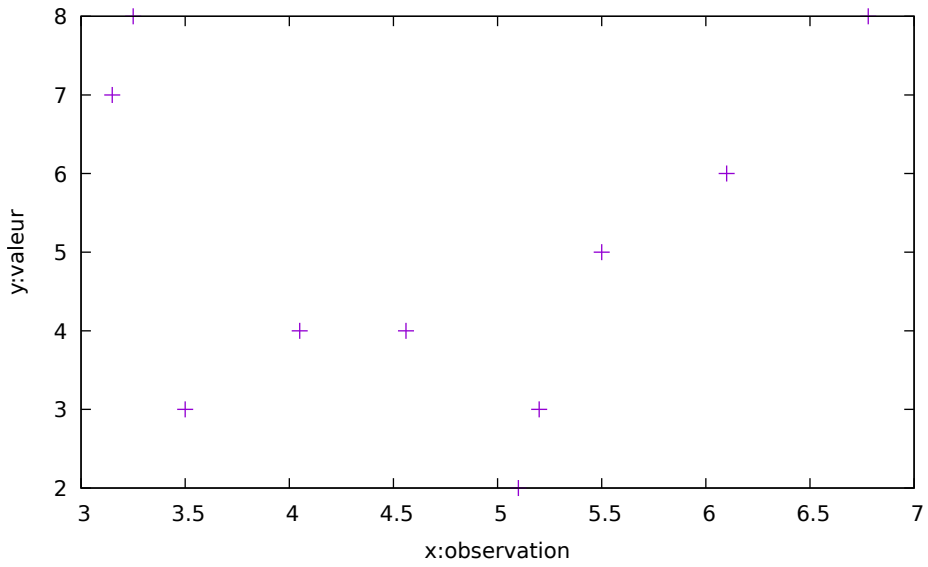


FIGURE 2.2 – Exemple d'observation

On définira m comme étant le nombre de couples (x, y) dans l'observation et n la dimension du descripteur(ici (2.2) $n = 1$).

1. non présente dans le vecteur de descripteurs.

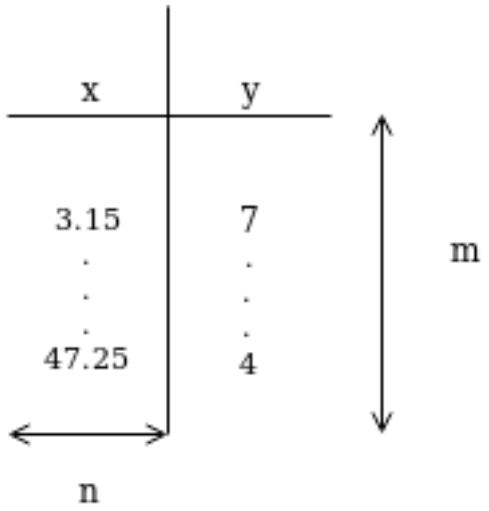


FIGURE 2.3 – Relation entre n,m,x et y

2.2 Régression linéaire

2.2.1 F, fonction de régression linéaire

A priori, F est une fonction linéaire dans un repère ou un plan [2].

2. On peut généraliser en parlant d'hyperplan.

2.2.2 Recherche de l'équation de la droite

Il est assez difficile de trouver une droite représentative de manière graphique.

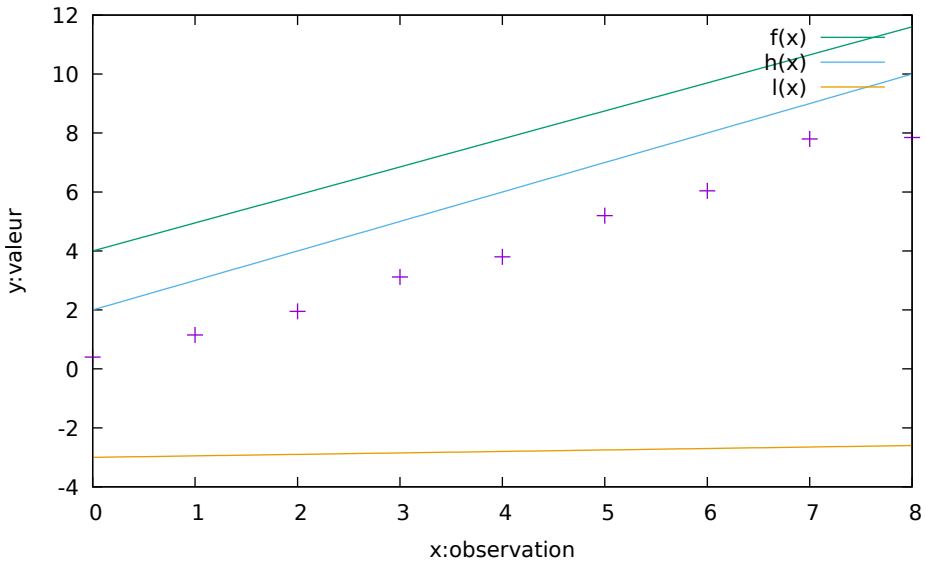


FIGURE 2.4 – Plusieurs fonction de regression peu satisfaisantes.

Il faut trouver la fonction F de manière algébrique :

$$F : y = a.\overrightarrow{x} + b \quad y, b \in \mathbb{R}, \overrightarrow{x} \in \mathbb{R}^n, a.\overrightarrow{x} \in \mathbb{R}$$

On remarque que $a.\overrightarrow{x} \in \mathbb{R}$, on en déduit que $a \in \mathbb{R}^n$ et que il y a un produit scalaire entre a et $\overrightarrow{x}$. Il y a cependant un [3] différence entre a et $\overrightarrow{x}$ pour que ce produit soit possible : a est un vecteur ligne et $\overrightarrow{x}$ un vecteur ligne.

$$a = \begin{pmatrix} a_0 & . & . & . & a_{n-1} \end{pmatrix} \quad \overrightarrow{x} = \begin{pmatrix} x_0 \\ . \\ . \\ . \\ x_{n-1} \end{pmatrix}$$

a est en fait la norme de de la droite représentant F, la norme étant un vecteur qui est perpendiculaire à une droite. De ce fait a est un vecteur colonne transposé et on doit noter l'équation de la droite de la manière suivante :

$$Y : y = a^T.\overrightarrow{x} + b$$

On verra à quoi correspond a^T par la suite.

3. on verra en fait que a est un vecteur colonne transposé.

2.2.3 Petits rappels sur les vecteurs

Produit scalaire

On définit le produit scalaire de la manière suivante :

$$a.\overrightarrow{x} = \sum_{i=0}^{n} a_i.x_i$$

Transposition

La transposition d'un vecteur ligne en fait un vecteur colonne et vice versa. On note v^T la transposé du vecteur v. Ainsi :

$$\overrightarrow{c}^T = \begin{pmatrix} c_0 & . & . & . & c_{n-1} \end{pmatrix}$$

$$l^T = \begin{pmatrix} a_0 \\ . \\ . \\ . \\ l_{n-1} \end{pmatrix}$$

c un vecteur colonne et l un vecteur ligne.
On pourra remarquer une propriété intéressante :

$$v^{T^T} = (v^T)^T = v$$

Du coup on peut écrire :

$$(a^T.\overrightarrow{x})^T = \overrightarrow{x}^T.a^{T^T} = \overrightarrow{x}^T.a$$

On rappelle aussi la chose suivante :

$$v.v^T = \|v\|^2$$

Le produit scalaire d'un vecteur et de sa transposé est égal à la norme [4] de ce vecteur au carré.

4. longeur

2.2.4 Forme général de F

On donne comme forme générale de F :

$$F : \Theta_1 . \overrightarrow{x} + \Theta_0$$

On va chercher à trouver Θ_1 et $Theta_0$ sachant que :
- Θ_0 est la valeur à l'origine de F.
- Θ_1 est le vecteur normal à la droite F.

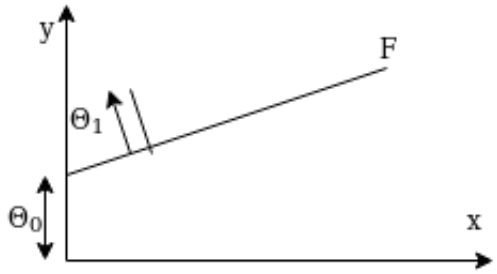


FIGURE 2.5 – Θ_0 et Θ_0 par rapport à F

Pour se faciliter la tâche on va réunir Θ_0 etΘ_1 dans un vecteur $\Theta.\overrightarrow{x'}$.

$$F : \Theta.\overrightarrow{x'}$$

Comment passer de x à x' ?

Prenons des valeurs en exemple :

$$x_1 = 3.3 \rightarrow x'_1 = \begin{pmatrix} 1 \\ 3.3 \end{pmatrix}$$

$$x_2 = 4.4 \rightarrow x'_2 = \begin{pmatrix} 1 \\ 4.4 \end{pmatrix}$$

$$x_n = 7.4 \rightarrow x'_n = \begin{pmatrix} 1 \\ 7.4 \end{pmatrix}$$

On a donc :

$$x' = \begin{pmatrix} 1 \\ x \end{pmatrix}$$

et

$$\Theta = \begin{pmatrix} \Theta_0 & \Theta_1 \end{pmatrix}$$

Pour calculer/apprendre Θ il est crucial de **prendre en compte l'observation**, afin de ne pas se retrouver dans la situation de la figure 2.4.

2.2.5 Fonction $J(\Theta)$

Prendre en compte les observations

On sait que $F(x_i) \rightarrow y_i$ est une fonction prédisant les valeurs de y. Pour prendre en compte les observation on va définir la fonction suivante :

$$J(\Theta) = \sum_{m}^{i=1} (F(x_i) - y_i)^2$$

que l'on pourra aussi écrire :

$$J(\Theta) = \sum_{m}^{i=1} (\Theta x_i' - y_i)^2$$

On notera que si $x \in \mathbb{R}^n$ alors $\Theta \in \mathbb{R}^{n+1}$
Analysons cette fonction :

- m correspond au nombre d'échantillons observés.
- Les valeurs que l'on cherche doivent être positives, d'où l'élevation au carré.
- $F(x_i)$ correspond aux valeurs prédites.
- y_i correspond aux valeurs observées.

Il va falloir travailler sur cette fonction $J(\Theta)$

Minimisation de $J(\Theta)$

Reprenons le processus de régression depuis le début :

1. On part d'un observation :$\{(features, classe)\}$
2. On connait la forme de la fonction droite.
3. $\Theta_0, \Theta_1 \rightarrow \Theta \begin{pmatrix} \Theta_0 & \Theta_1 \end{pmatrix}$ et $x \rightarrow x' \begin{pmatrix} 1 \\ x \end{pmatrix}$.
4. On définit la fonction que l'on cherche minimiser :

$$J(\Theta) = \sum_{m}^{i=1} (\Theta x_i' - y_i)^2$$

.

Minimiser une fonction revient à trouver le minimum de la fonction. Donc minimiser une fonction f(x) revient à résoudre f(x)=0.
Pour $J(\Theta)$ on trouve une solution unique quand $m = n + 1$, mais la plupart du temps il va falloir approximer la solution.

2.2.6 Dérivé partielle et Gradient

Dérivée partielle

Pour trouver le minimum il va falloir utiliser la notion de **dérivée partielle de fonction**. Par exemple la dérivé partielle de $X.x_0 + Y.x_1$ en x_0 puis en x_1.

$$\frac{\partial(X.x_0 + Y.x_1)}{\partial x_0} = X \quad \frac{\partial(X.x_0 + Y.x_1)}{\partial x_1} = Y$$

Pour notre fonction $J(\Theta)$:

$$\frac{\partial J(\Theta)}{\partial \Theta_0} \quad \frac{\partial J(\Theta)}{\partial \Theta_1}$$

Gradient

On appelle gradient d'un fonction $f(x) = X.x_0 + Y.x_1$ la chose suivante :

$$\overrightarrow{\bigtriangledown} f(x) = \begin{pmatrix} \frac{\partial(X.x_0+Y.x_1)}{\partial x_0} \\ \frac{\partial(X.x_0+Y.x_1)}{\partial x_1} \end{pmatrix}$$

Retranscrit à la fonction $J(\Theta)$ on obtient :

$$\overrightarrow{\bigtriangledown} J(\Theta) = \begin{pmatrix} \frac{\partial J(\Theta)}{\partial \Theta_0} \\ \frac{\partial J(\Theta)}{\partial \Theta_1} \end{pmatrix}$$

2.2.7 Que cherche-t-on à résoudre ?

Le but de ce calcul est de trouver 1 ou plusieurs Θ tel que $\overrightarrow{\bigtriangledown} J(\Theta) = \overrightarrow{0}$.
Il y a deux solution :

La solution analytique

Il faut donc résoudre le système suivant :

$$\frac{\partial J(\Theta)}{\partial \Theta_0} = 0 \quad \frac{\partial J(\Theta)}{\partial \Theta_1} = 0$$

Cependant réussir à calculer la dérivée de cette fonction est compliquer rendant cette solution souvent impossible à appliquer.

La solution algorithmique la descente de gradient

.

$$\frac{\partial J(\Theta)}{\partial \Theta_0} \quad \frac{\partial J(\Theta)}{\partial \Theta_1}$$

Il faut de prime abord comprendre que cette méthode ne fonctionne pas toujours pour plusieurs raisons que nous allons identifier.
Le principe de cet algorithme est le suivant : **Aller dans le sens de la pente**.

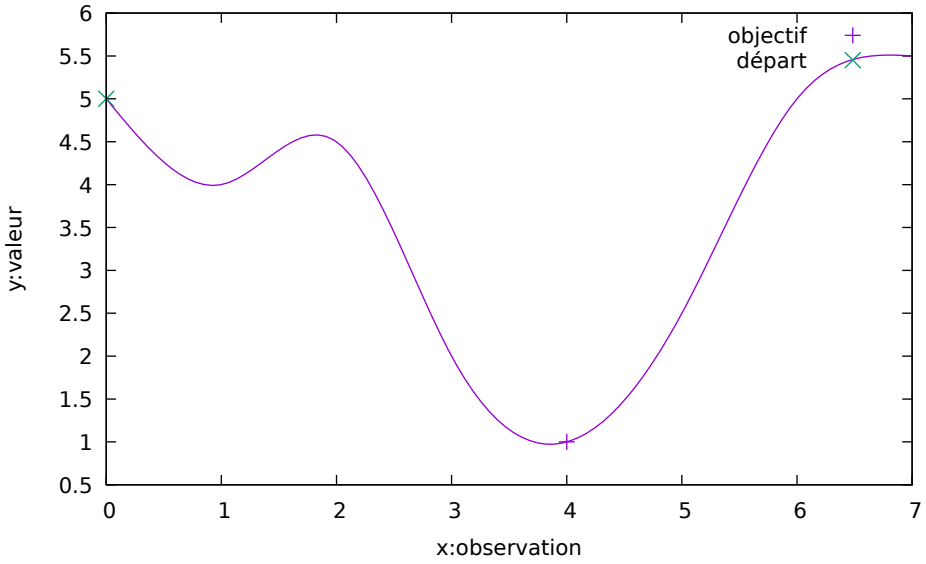


FIGURE 2.6 – Principe de la descente de gradient

L'aide de la dérivé de la fonction, on va aller vers la gauche si f est décroissante ou aller vers la droite si f est croissante.
On se déplace dans le sens inverse du signe de la dérivée. On peut tout de suite remarquer qu'il peut y avoir des "faux" minimums et que si un fonction est convex sur une intervalle on peut se retrouver à osciller indéfiniement.

2.2.8 Descente de gradient

Trouver la formule pour la solution

La solution se calcul ainsi : Solution à n+1 = solution à n-$\overrightarrow{\bigtriangledown} J()$ à l'étape n. On peut donc donner cette formule :

$$\Theta_{n+1} = \Theta_n - \overrightarrow{\bigtriangledown} J(\Theta_n)$$

On ne s'arrête que lorsque $\overrightarrow{\bigtriangledown} J(\Theta) \approx \overrightarrow{0}$, c'est à dire tant que $\|J(\Theta)^{n+1}\| \geq \epsilon$
Cependant il y a de nombreux problèmes :

Faux minimum

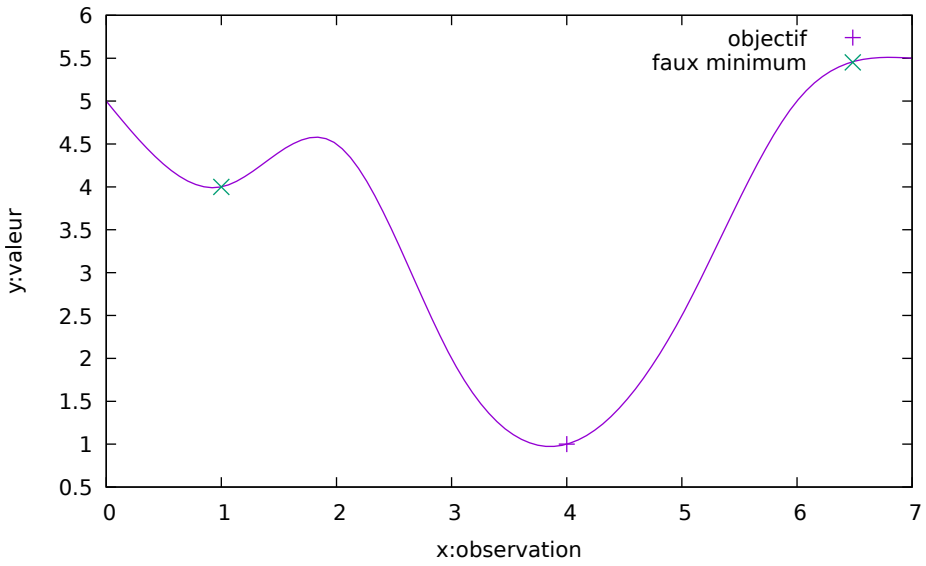


FIGURE 2.7 – Exemple de faux minimum

Le minimum réél n'est jamais atteint.

Arrêt de descente

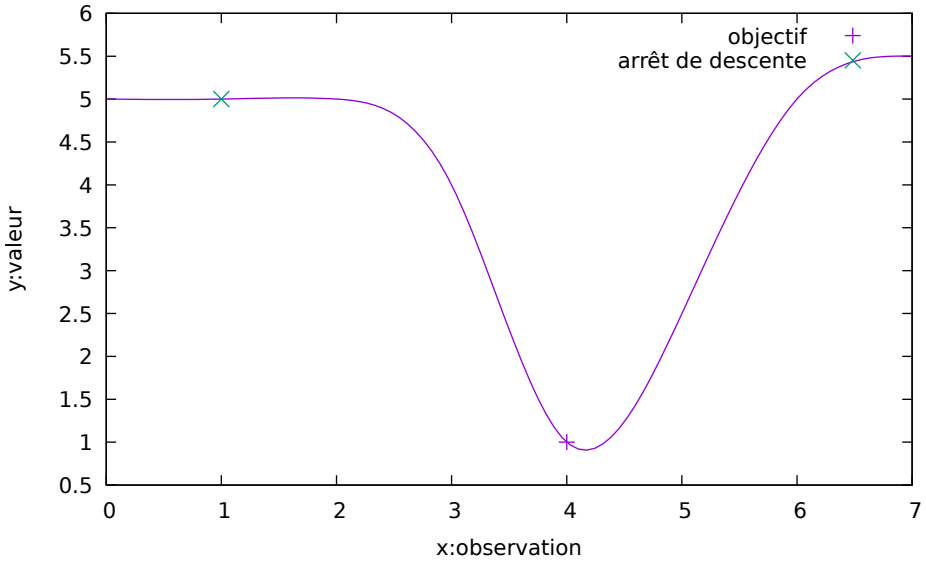


FIGURE 2.8 – Exemple d'arrêt de descente

Le minimum n'est pas garanti si la courble est localement constante.

Pente trop douce

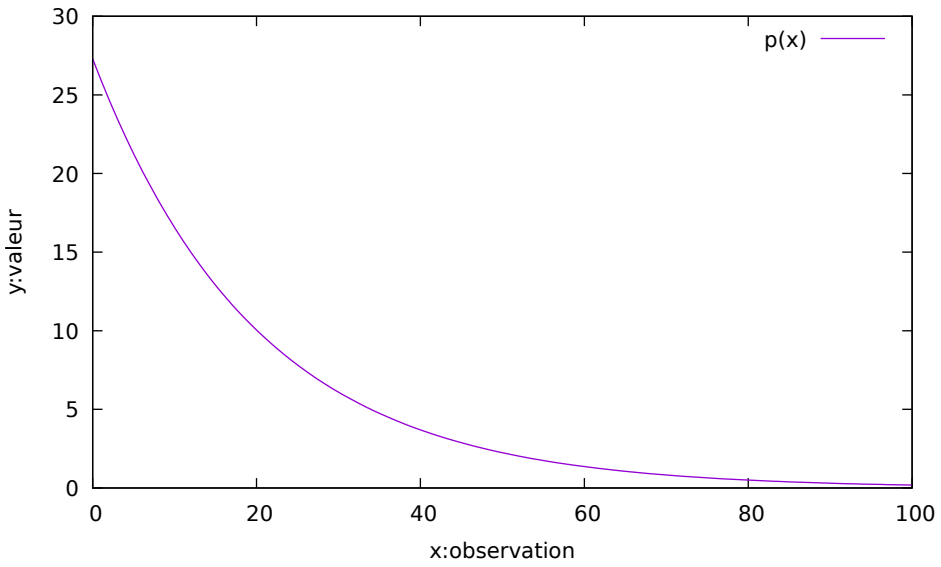


FIGURE 2.9 – Exemple de pente trop douce

Ici le problème vient du fait que plus on avance sur la pente plus $\overrightarrow{\bigtriangledown} J(\Theta)$ est trop petit, la descente s'arrêta à quand ϵ sera atteint et ce bien avant le réel résultat.

Solution

Afin de résoudre ce problème on fait appel à **un coefficient d'amortissement** noté $\eta(n)$.

$$\eta(n) = 1 \quad \text{Descente constante}$$
$$\eta(n) > 1 \quad \text{Descente accélérée}$$
$$\eta(n) < 1 \quad \text{Descente amortie}$$
$$\eta(n) = 0 \quad \text{Pas de descente}$$

Ainsi on cherche le **Gradient amorti** :

$$\Theta_{n+1} = \Theta_n - \eta(n).\overrightarrow{\bigtriangledown} J(\Theta_n)$$

Remarques

Θ_0 est initialisé aléatoirement ou de manière choisie Le calcul de gradient est un calcul long, on va donc vouloir se concentrer dès le départ sur une petite portion choisie du gradient.

Chapitre 3

Descente de gradient

3.1 Rappels

3.1.1 Processus itératif

La descente de gradient s'effectue sur une fonction $J(\theta)$, une fonction objective permettant de prédire une valeur en fonction des descripteurs fournis. On va donc chercher à trouver θ tel que $J(\theta)$ soit **minimal**.
Pour cela on utilise le procédé itératif suivant :

$$\text{Tant que } (\theta_{k+1} - \theta_k) \geq \epsilon$$

$$\Theta_{k+1} = \Theta_k - \eta(k).\overrightarrow{\bigtriangledown} J(\Theta_k)$$

On rappelle que $\eta(k)$ est un coefficient d'amortissement/d'accélération(en fonction de sa valeur) et que $\overrightarrow{\bigtriangledown} J(\theta)$ est le gradient de la fonction $J(\theta)$ tel que :

$$\overrightarrow{\bigtriangledown} J(\theta) = \begin{pmatrix} \frac{\partial J(\theta)}{\partial \theta_0} \\ ... \\ \frac{\partial J(\theta)}{\partial \theta_i} \end{pmatrix}$$

La descebte va dans le sens inverse du signe de la dérivée.

3.1.2 Regression linéaire

On réalise une régression à partir des obersvations :

obervation = (x_i, y_i),x_i est un descripeur et y_i la valeur observée *Exemple* : $((3, 2), 4)$, $x = (3, 2)$ et $y = 4$.

On peut définir une regression linéaire et ainsi définir la forme générale de la fonction de regression $h(x)$:

$$h(x) = \theta x + \theta_0$$

θ est de dimensions $\mathbb{R}^n$.

Remarque sur les hyperplans

La définition d'un hyperplan est la suivante :

Un espace vectoriel de de dimension $n-1$ si la dimension est de n

En clair si on est dans espace de dimension :
- 2 (x, y) : alors l'hyperlan sera une droite
- 3 (x, y, z) : alors l'hyperplan sera un plan
- etc..

3.1.3 Regression linéraire (suite)

On cherche ensuite à simplifier la fonction $h(x)$. Pour cela on défini :

$$\theta" = \begin{pmatrix} \theta_0 \\ \theta \end{pmatrix}$$

De ce fait on re-défini x en $x"$:

$$x_i" = \begin{pmatrix} 1 x_i \end{pmatrix}$$

On a donc maintenant pour fonction :

$$h(x") = \theta" x"$$

Et on définit la fonction $J(\theta")$:

$\sum_{i=1}^{m} (\theta" x_i" - y_i)^2$, m étant le nombre d'observation et y la valeur observée

C'est à partir de cette écriture qu'on a pu définir $\overrightarrow{\bigtriangledown} J(\Theta)$:

$$\overrightarrow{\bigtriangledown} J(\Theta_k) = \begin{pmatrix} \frac{\partial J(\theta)}{\partial \theta_0} \\ ... \\ \frac{\partial J(\theta)}{\partial \theta_n} \end{pmatrix}$$ n étant la dimension des descripteur

3.2 Dérivée de la fonction $J(\theta)$

On va maintenant voir comment calculer et noter la dérivée de la fonction $J(\theta")$:

$$\frac{\partial \sum_{i=1}^{m}(\theta" x_i" - y_i)^2}{\partial \theta_i} = \frac{\partial \sum_{i=1}^{m}((\theta_0 \quad ... \quad \theta_m)\begin{pmatrix}1\\x_{i,1}\\...\\x_{i,m}\end{pmatrix} - y_i)^2}{\partial \theta_i}$$

$$= \frac{\partial \sum_{i=1}^{m}((\theta_0 + \theta_1 x_{i,1} + ... + \theta_n x_{i,n}) - y_i)^2}{\partial \theta_i}$$

On notera la chose suivante :

$$x_{i,k} = \frac{\partial(\theta_0 + ... + \theta_n x_{i,n}) - y_i}{\partial \theta_i}$$

Ainsi on peut donner la formule suivante :

$$\frac{\partial J(\theta)}{\theta_k} = 2\sum_{i=1}^{m} x_{i,k}(\theta" x_i" - y_i)$$

Comme on cherche à minimiser la forumle peut être réécrite ainsi :

$$\frac{\partial J(\theta)}{\theta_k} = \sum_{i=1}^{m} x_{i,k}(\theta" x_i" - y_i)$$

en effet le coefficient 2 influe peu sur le résultat. On peut donc donner $\overrightarrow{\bigtriangledown} J(\Theta_k)$ sous cette forme

$$\overrightarrow{\bigtriangledown} J(\Theta) = \begin{pmatrix}\sum_{i=1}^{m} x_{i,0}(\theta" x_i" - y_i)\\...\\\sum_{i=1}^{m} x_{i,k}(\theta" x_i" - y_i)\\...\end{pmatrix} \quad \text{avec} \forall j, x_{0,j} = 0$$

Exemple

On possède les obersvation suivantes :

$$((2,3),1) \quad ((4,5),3)$$

On cacule le $\overrightarrow{\bigtriangledown} J(\Theta)$ à partir des ces observations :

$$\overrightarrow{\bigtriangledown} J(\Theta) = \begin{pmatrix} 1 \times \theta \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} - 1 + 1 \times \theta \begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix} - 3 \\ 2 \times \theta \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} - 1 + 4 \times \theta \begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix} - 3 \\ 3 \times \theta \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} - 1 + 5 \times \theta \begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix} - 3 \end{pmatrix}$$

Maintenant que l'on sait calculer le gradient on peut appliquer le processus itératif pour trouver un minimum de $J(\theta)$

Remarque du professeur

Savoir appliquer la descente de gradient ainsi sur un exemple est demandé à l'examen

3.3 Reformulation sous forme matricielle

Exemple de mise en forme

Soit l'observation suivante :

$$Obs = \{((2,3),1),((4,5),3),((5,4),8)\}$$

3.3.1 Forme Matricielle de $J(\theta)$

On définira les matrices et vecteurs suivants :

$$X = \begin{vmatrix} 1 & 2 & 3 \\ 1 & 4 & 5 \\ 1 & 5 & 4 \end{vmatrix} \quad \overrightarrow{y} = \begin{vmatrix} 1 \\ 3 \\ 8 \end{vmatrix} \quad \Theta" = \begin{vmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{vmatrix}$$

On peut faire le lien suivant :

$$J(\theta") = \sum_{i=1}^{m} (\theta" x_i" - y_i)^2 = (X\Theta" - \overrightarrow{y})(X\Theta" - \overrightarrow{y})^T$$

$(X\Theta" - \overrightarrow{y})$ est un vecteur colonne

$(X\Theta" - \overrightarrow{y})^T$ est un vecteur ligne

$J(\theta")$ est donc un scalaire($\in \mathbb{R}$)

3.3.2 Dérivé de la forme matricielle

On va très rapidement donner les étapes de la dérivation de la forme matricielle :

$$\frac{\partial J(\Theta")}{\partial \Theta"} = (X\Theta" - \overrightarrow{y})(X\Theta" - \overrightarrow{y})^T$$

$$= \frac{\partial (X\Theta")^T X\Theta" - (X\Theta")^T \overrightarrow{y} - \overrightarrow{y}^T X\Theta" + \overrightarrow{y}^T \overrightarrow{y}}{\partial \Theta"}$$

$$= \frac{\partial (X\Theta")^T X\Theta" - 2\overrightarrow{y}^T (X\Theta")}{\partial \Theta"}$$

$$\frac{\partial J(\Theta")}{\partial \Theta"} = XX^T\Theta" - X^T\overrightarrow{y}$$

Ainsi lorsque l'on veut résoudre

$$\frac{\partial J(\Theta")}{\partial \Theta"} = 0$$

Il suffit de résoudre :

$$XX^T\Theta = X^T\overrightarrow{y}$$

Remarque : le résultat d'un produit matriciel entre une matrice et sa transposé est une matrice carré. XX^T est **une matrice carré** Si la matrice XX^T est inversible alors on peut calculer $(XX^T)^{-1}$ et donc résoudre l'équation suivante [1] :

$$\Theta" = (XX^T)^{-1}X^T\overrightarrow{y}$$

3.3.3 Problèmes de la forme matricielle

Est-ce qu'il y a forcément une solution à l'équation suivante ? :

$$X\Theta" - \overrightarrow{y} = \overrightarrow{0} \quad (X \in (\mathbb{R}^n \times \mathbb{R}^m), \Theta \in \mathbb{R}^n, \overrightarrow{y} \in \mathbb{R}^m)$$

On noter a qu'il y aura 10^6 avec seulement 3 inconnue, et qu'une inversionde matrice à une complexité en temps de $O(n^3)$.
La réponse est non. La solution va uniquement dépendre de l'ordre de grandeur des informations prises. Une solution est de choisir des sous-échantillons.

3.3.4 À retenir

— D'une observation on obtient θ.
— Méthode directe : $\Theta = (XX^T)^{-1}X^T\overrightarrow{y}$
— Méthode indirecte : descente de gradient par processus itératif.

1. on peut obtenir ce résultat sur mathlab en écrivant $\Theta" = y\backslash X$

Chapitre 4

Bayesienne

4.0.1 Principe

On connait la forme de la distribution. On ne **connait pas les paramètres** mais on connait la **la loi de probabilité**

Exemple

Problèmes à deux classes : "Bar" et "Saumon".
On possède la vérité terrain suivante :

$$VT = \begin{matrix} (Feature_1, "Bar") \\ (Feature_2, "Saumon) \\ ... \\ (Feature_n, "Bar) \end{matrix}$$

On doit prendre une décision en fonction de la Vérité Terrain et avec l'aide d'une loi de distribution de X(une loi de probabilité), en fonction de "Bar" ou "Saumon"

4.1 Différentes approches

4.1.1 Maximum à priori

La bonne démarche est la suivante :

$$VT = \begin{matrix} (\emptyset, "w_1") \\ ... \\ (\emptyset, "w_2) \end{matrix}$$

On connait le nombre de "Bar": n_{w_1} et le nombre de "Saumon" n_{w_2}
On donne les probabilités suivante :

$$P(w_1) = \frac{n_{w_1}}{|VT|}$$

$$P(w_2) = \frac{n_{w_2}}{|VT|}$$

Pour vérifier que il s'agisse bien d'une loi de probabilité on s'assure que :

$$\frac{n_{w_1} + n_{w_2}}{|VT|} = 1$$

On va donc pouvoir dire si $P(w_1) > P(w_2)$ alors "w_1", sinon "w_2".
Ce classifieur ne prend pas en compte les features.

4.1.2 Maximum vraisemblable

Pour ce classifieur on va étudier les features en fonction de l'étiquette de la classe.
Notre Vérité Terrain devient :

$$VT = \begin{matrix} (f_1, "w_1") \\ ... \\ (f_n, "w_2") \end{matrix}$$

Il nous faudra donc définir : $P(x = f|w)$ la probabilité que f est la valeur x sachant qu'il appartient à la classe w.
Pour x donné il faut décider s'il appartient à "w_1" ou "w_2" :

$$P(x = f|w_1)$$

$$P(x = f|w_2)$$

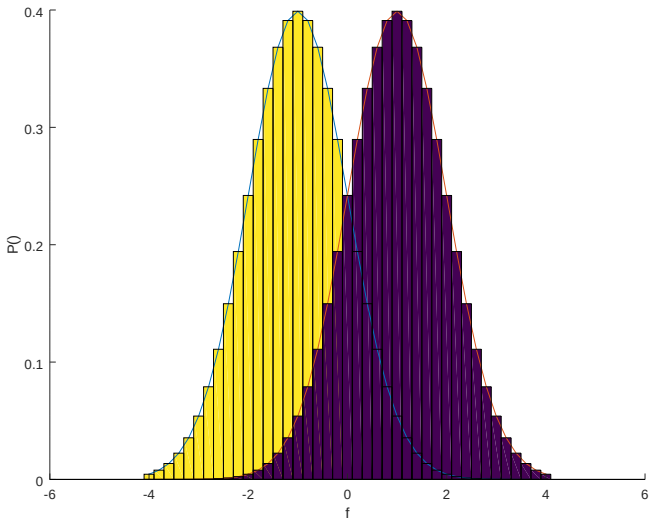


FIGURE 4.1 – Maximum vraisemblable

On passe de données discrètes à des données continue suivant une certaine loi de probabilité. Ici on suit une loi normale $\mathcal{N}(\mu_1, \sigma_1)$ et $\mathcal{N}(\mu_2, \sigma_2)$

Récapitulatif 1

1. On part d'une Vérité Terrain et on identifie 2 classes : w_1,w_2
2. On identifie (μ_1, σ_1),(μ_2, σ_2)
3. On calcule $P(x = f|w_1)$ et $P(x = f|w_2)$

$$P(x = f|w) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

$$P(x = f|w_1) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2}$$
$$P(x = f|w_2) = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2}$$

Fonctionnement

Ce classifieur prend x et répond w_1 ou w_2.
On va donc comparer $P(x = f|w_1)$ $P(x = f|w_2)$:

$$\text{Si } P(x = f|w_1) > P(x = f|w_2) \text{ alors } w_1 \text{ sinon } w_2$$

Probabilité d'erreur et seuil d'erreur

$P(erreur) = max[P(x = f|w_1), P(x = f|w_2)]$, $\int_{f_1}^{f_n} P(erreur)dx = 1$
Pour trouver le seuil d'erreur on va résoudre l'équation suivante :

$$P(x = f|w_1) = P(x = f|w_2)$$
$$\frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2} = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2}$$
$$\log(\frac{1}{\sigma_1} e^{-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2}) = \log(\frac{1}{\sigma_2} e^{-\frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2})$$
$$-\frac{1}{2}(\frac{x-\mu_1}{\sigma_1})^2 + \frac{1}{2}(\frac{x-\mu_2}{\sigma_2})^2 = -\log\sigma_1 + \log\sigma_2$$

L'erreur correspondra à la surface créée par le croisement des deux courbes gaussiennes.
On obtient une équation du second degré qui est résolvable assez facilement.
Ensuite on pourra réaliser la comparaison suivante :

$$\text{Si } x > Z \text{ alors } w_1 \text{ sinon } w_2$$

On notera qu'il y a plusieurs seuils comparables, un par descripteur. On devra donc réaliser la suite de comparaison suivante :

$$\begin{array}{r} \text{si } x < z_1 \text{ alors } w_1 \\ \text{sinon si } x < z_2 \text{ alors } w_2 \\ \dots \\ \text{sinon } w_1 \end{array}$$

Récapitulatif 1

1. On part d'une Vérité Terrain et on identifie 2 classes : w_1,w_2
2. On identifie (μ_1, σ_1),(μ_2, σ_2)
3. On calcule $P(x = f|w_1)$ et $P(x = f|w_2)$

Récapitulatif 2

1. On part d'une Vérité Terrain et on identifie 2 classes : w_1,w_2
2. On identifie (μ_1, σ_1),(μ_2, σ_2)
3. On calcule $P(x = f|w_1)$ et $P(x = f|w_2)$
4. On résout l'équation $P(x = f|w_1) = P(x = f|w_2)$
5. On fait une comparaison sur chaque z_i de chaque descripteur ;

4.1.3 Maximum à posteriori

Pour ce type de classifieur on utilisera la formule d Bayles :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Dans le cadre du classifieur on va chercher à donner la formule $P(w|x = f)$;

$$P(w|x = f) = \frac{P(x = f|w)P(w)}{P(x = f)}$$

Si l'on reprend le précédent classifieur, on se sert de la distribution des features dans la classe.
Ici, on se sert des features pour trouver la classe. On pourra réaliser la comparaison suivante :

$$P(w_1|f = x) > P(w_2|f = x)$$
$$\frac{P(f = x|w_1)P(w_1)}{P(f = x)} > \frac{P(f = x|w_2)P(w_2)}{P(f = x)}$$

$P(f = x)$ n'influançant pas la comparaison, on peut la simplifier .

$$P(f = x|w_1)P(w_1) > P(f = x|w_2)P(w_2)$$

Ainsi :

$$\text{Si } P(f = x|w_1)P(w_1) > P(f = x|w_2)P(w_2 \text{ alors } w_1 \text{ sinon } w_2$$

4.2 Loi normal en dimension N

On s'intéresse désormais aux descripteurs de dimension $\mathbb{R}^n$

4.2.1 Fonction gaussienne dans $\mathbb{R}^n$

$$P(x) = \frac{1}{(2\pi)^{1/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

Avec :
— Σ une matrice de covarience $\in \mathbb{R}^n$
— Σ^{-1}, l'inverse Σ
— μ le vecteur moyen de l'échantillon $\in \mathbb{R}^n$
— $|\Sigma|$ le déterminant de la matrice Σ,$\in \mathbb{R}$

4.2.2 Matrice de covariance

La matrice de covariance est une matrice carré, symétrique, positive ayant la forme suivante :

$$\begin{bmatrix} \sigma_{1,1}^2 & \sigma_{1,2} & \dots & \sigma_{1,n} \\ \sigma_{2,1} & \sigma_{2,2}^2 & & \\ \dots & & \dots & \\ \sigma_{n,1} & & & \sigma_{n,n}^2 \end{bmatrix}$$

On a les formules suivantes [1] :

$$\sigma_{i,i}^2 = \frac{1}{k} \sum_{l=1}^{l=k} (x_{l,i} - \mu_i)^2$$

et :

$$\sigma_{i,j}^2 = \frac{1}{k} \sum_{l=1}^{l=k} (x_{l,i} - \mu_i)(x_{l,j} - \mu_j)$$

On a aussi :

$x_{l,i}, l$: le l-ième descripteur, i : le i-ème élement du descripteur mail

$$\mu = \begin{bmatrix} \mu_1 \\ \dots \\ \mu_j \\ \dots \\ \mu_n \end{bmatrix} = \begin{bmatrix} \frac{1}{n} \sum_1^l x_{l,1} \\ \dots \\ \frac{1}{n} \sum_1^l x_{l,j} \\ \dots \\ \frac{1}{n} \sum_1^l x_{l,n} \end{bmatrix}$$

D'ailleurs on notera que $\sigma_{i,j} = \sigma_{j,i}$.

Exemple

On possède les descripteurs suivants :

$$(1,4), (2,5), (3,9)$$

On aura la matrice suivante :

$$\begin{bmatrix} \sigma_{1,1}^2 & \sigma_{1,2} \\ \sigma_{2,1} & \sigma_{2,2}^2 \end{bmatrix}$$

On calcule donc les valeurs suivantes :

$$\mu_1 = \frac{1+2+3}{3} = 2$$

$$\mu_2 = \frac{4+5+9}{3} = 6$$

1. $\sigma_{i,i}^2$ est une notation pour signifier que ce σ précis est sur la diagonale.

$$\sigma_{1,1}^2 = \frac{(1-2)^2 + (2-2)^2 + (3-2)^2)}{3} = \frac{2}{3}$$
$$\sigma_{2,2}^2 = \frac{(4-6)^2 + (5-6)^2 + (9-6)^2)}{3} = \frac{14}{3}$$
$$\sigma_{1,2}^2 = \frac{(1-2)(4-6) + (2-2)(5-6) + (3-2)(9-6)}{3} = \frac{5}{3}$$

Ainsi on obtient :

$$\begin{bmatrix} \frac{2}{3} & \frac{5}{3} \\ \frac{5}{3} & \frac{14}{3} \end{bmatrix}$$

4.3 $\mathcal{N}(\mu, \sigma)$ en dimension 2

4.3.1 Distance

Si l'on regarde la fonction gaussienne on voit apparaître :

$\frac{x-\mu}{\sigma}$, une formule de distance

En dimension 2 on peut considérer les identités suivantes :

$$\Sigma = \sigma^2$$
$$\Sigma^{-1} = \frac{1}{\sigma^2}$$

Formule de la distance de Mahalanobis

On donne la formule suivante :

$$\sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

Cette formule permet de remplace la distance euclidienne car elle prend en compte la matrice Σ^{-1}

4.3.2 Courble d'iso probabilité

Il est possible de tracer des courbes d'isoprobabilité [2] à partir des matrices de covariance Σ. On distinguera plusieurs cas particuliers :

2. ellipses concentriques. Chaque ellipse représente unne probabilité et tout point appartenant à la même ellipse ont la même probabilité.

Courbes d'iso-probabilité : cercles

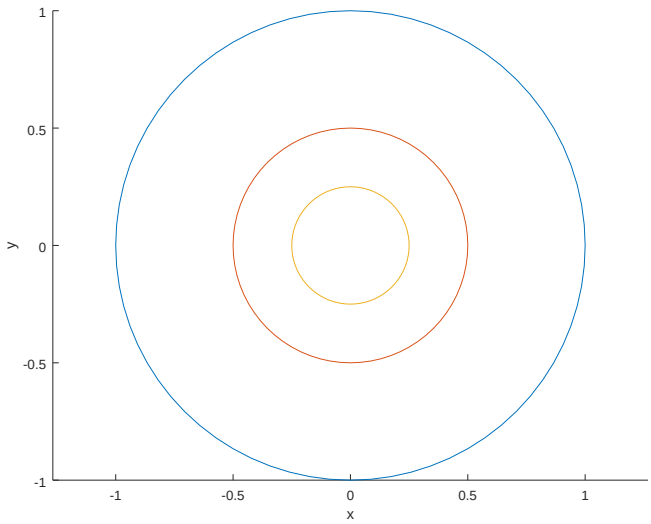


FIGURE 4.2 – Courbes d'isprobabilitiés

Dans ce cas particulier la matrice Σ est de la forme suivante :

$$\Sigma = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix} = \alpha * Identity$$

On remarquera que si Σ est inversible alors :

$$\Sigma^{-1} = \frac{1}{\alpha} * Identity$$

Courbes d'iso-probabilité : ellipses parrallèles aux axes

Dans ce cas la matrice Σ est la suivante :

$$\Sigma = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$$

La distinction va se jouer entre α et β.

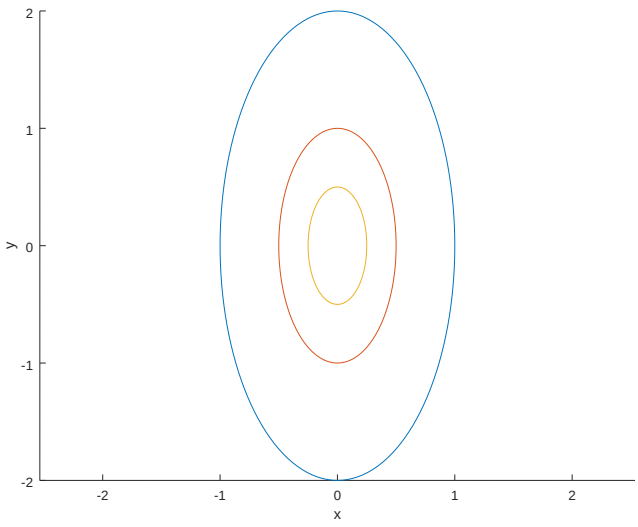


FIGURE 4.3 – Cas ou $\alpha < \beta$

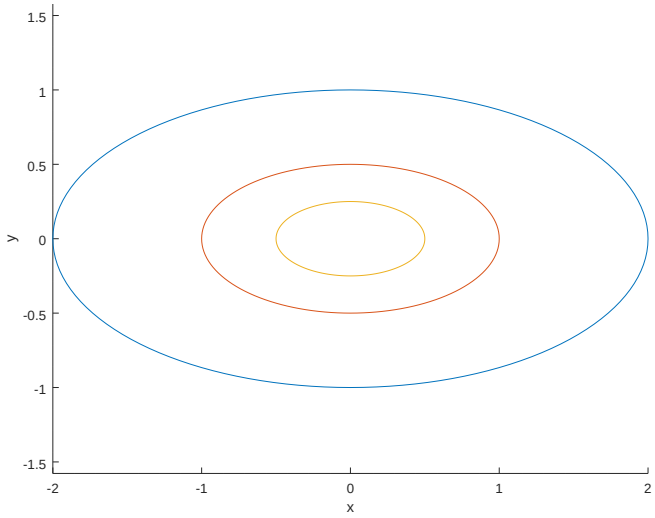


FIGURE 4.4 – Cas ou $\beta < \alpha$

Coubes d'iso-probabilité : ellipses non parallèles

Dans ce cas la matrice Σ sera similaire au cas précédent, si ce n'est que l'on introduit une variable γ :

$$\Sigma = \begin{bmatrix} \alpha & \gamma \\ \gamma & \beta \end{bmatrix}$$

Ainsi l'orientation ne dépendra plus que de α et β mais aussi de la valeur de γ.

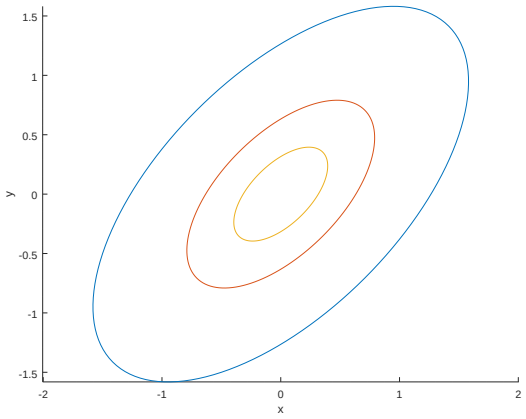


FIGURE 4.5 – $\gamma>0$ et $\beta>\alpha$

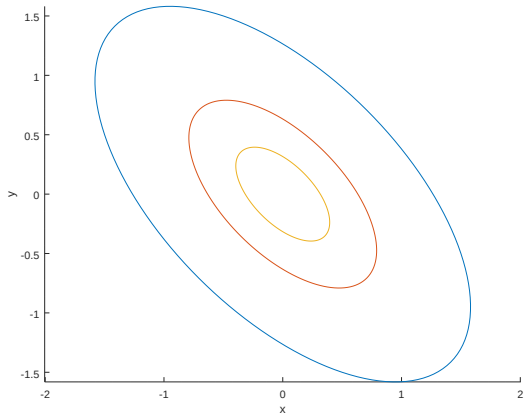


FIGURE 4.6 – $\gamma<0$ et $\alpha>\beta$

4.3.3 Forme de la frontière de séparation

Si l'on a plusieurs observation nous obtiendrons plusieurs courbes d'iso-probabilités liées. Il va donc falloir pouvoir délimiter ces courbes entre elles.

Cas $\Sigma_1 = \Sigma_2 = \alpha * Id$

Dans le cas où $\Sigma_1 = \Sigma_2 = \alpha * Identity$, deux lois normales de matrices de covariance $\Sigma = \alpha * Identity$, on cherche x tel que $P_1(x) = P_2(x)$. Les courbes d'iso-probabilités seront 2 cercles.
On sait que :

— $P(x|w_1)P(w_1)$ est une loi normale, multi-varié, avec $\Sigma = \alpha * Id$ et $\mu = \mu_1$
— $P(x|w_2)P(w_2)$ est une loi normal, multi-varié, avec $\Sigma = \alpha * Id$ et $\mu = \mu_2$

On souhaiterai résoudre :

$$P(x|w_1)P(w_1) = P(x|w_2)P(w_2)$$

Avec quelques calculs on peut arriver à une fonction de la forme :

$$w^T x + x_0 = 0$$

Avec :

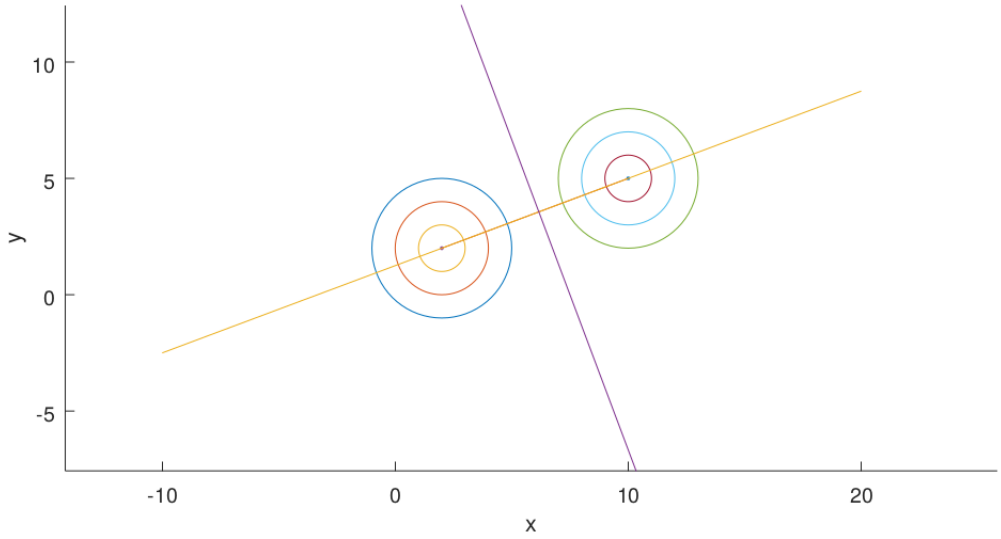
$$x_0 = -\log(\frac{P(w_2)}{P(w_1)}) + \frac{1}{2}\mu_1^T\mu_1 - \mu_2^T\mu_2$$

$$w^T = -\frac{1}{\alpha}(\mu_2^T - \mu_1^T)$$

On notera aussi :

$$w = -\frac{1}{\alpha}(\mu_2 - \mu_1)$$

La fonction est une droite orthogonale à la droite passant par le millieu (des cercles).

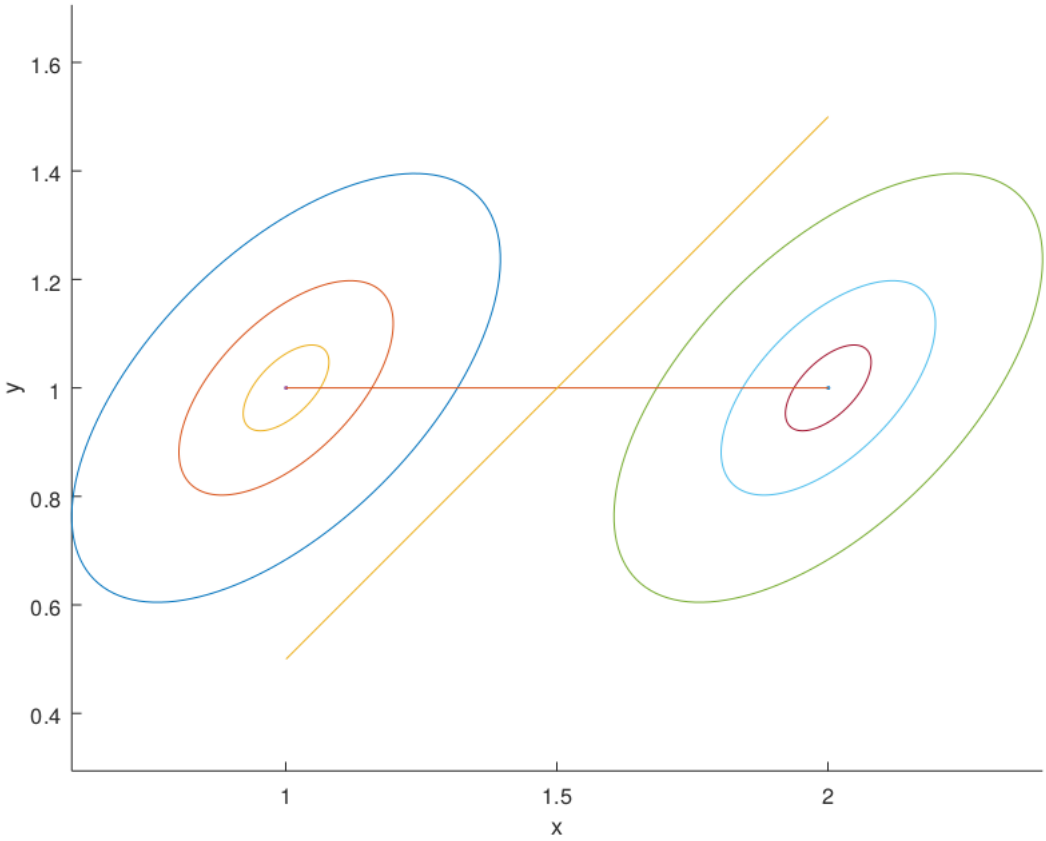


Cas $\Sigma_1 = \Sigma_2 \neq \alpha * Id$

Dans ce cas précis, les courbes sont deux ellipses parallèles par rapport au grand axe. La droite de séparation ne sera plus orthogonale à la droite passant par les millieu. La droit de séparation sera parallèle au grands axes des ellipses. On fait à nous ici face à une notion de distance (celle de Mahalanobis) :

$$(x - \mu_1)^T\Sigma^{-1}(x - \mu_1)$$

$$(x - \mu_2)^T\Sigma^{-1}(x - \mu_2)$$



Cas $\Sigma_1 \neq \Sigma_2$

Dans ce cas précis, la fonction de la droit de séparation sera une fonction quadratique de la forme suivante :

$$x^T wx + w^T + x_0$$

Remarque importante

Il est important de pouvoir expliquer l'origine de la forme de ces droites de séparation.

Chapitre 5

Réduction en dimension

5.1 Objectif

Le but de la réduction en dimension est de passer d'une dimension $\underline{\mathbb{R}^d}$ à une dimension $\underline{\mathbb{R}^k}$ tel que $\underline{k < d}$.
On va donc passer faire la transition d'un vecteur $\overline{x}$ de descripteur de dimension d à une vecteur $\overrightarrow{y}$de descripteur de dimension k. Pour ce faire il nous faudra trouver une matrice de transformation W.

$$\begin{bmatrix} x_{1,1} \\ \dots \\ x_{1,d} \end{bmatrix} \rightarrow^{W} \begin{bmatrix} y_{1,1} \\ \dots \\ y_{1,k} \end{bmatrix}$$

Sachant que l'opération à réaliser est la suivante :

$$\begin{bmatrix} x_{1,1} \\ \dots \\ x_{1,d} \end{bmatrix} = W \begin{bmatrix} y_{1,1} \\ \dots \\ y_{1,k} \end{bmatrix}$$ On peut déterminer que W est de la forme$W(k, d)$

Ainsi :

$$W = \begin{bmatrix} w_{1,1} & \dots & w_{1,d} \\ \dots & & \dots \\ w_{k,1} & \dots & w_{k,d} \end{bmatrix}$$

5.2 Avantages et inconvénients

La réduction en dimension va avoir plusieurs effets sur les observations qui se traduisent principalement par une compression de l'information.

5.2.1 Inconvénients

La compression d'information peut générer une perte d'information, en effet si on observe l'exemple suivant, la réduction de $\mathbb{R}^2$ à $\mathbb{R}^1$ provoque une perte notable d'information :

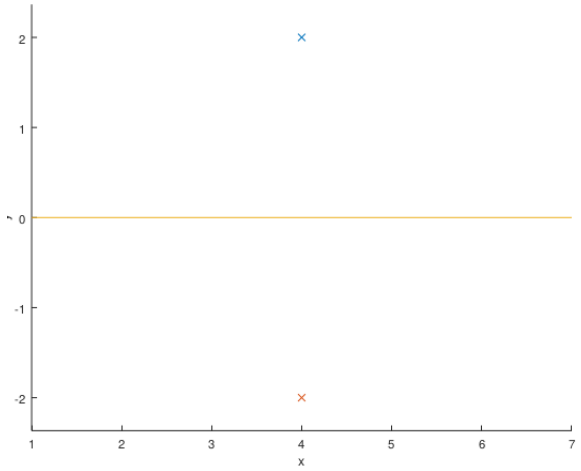


FIGURE 5.1 – Avant perte d'informations

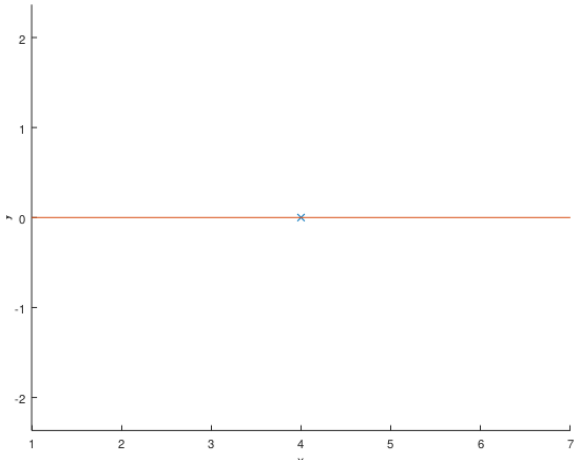


FIGURE 5.2 – Après perte d'informations

5.2.2 Avantages

La réduction va permettre de **densifier** l'information et donc provoquer une augmentation de la distance entre les échantillons (ce qui permet de trouver plus facilement une frontière). Par exemple si nous avons à faire à une observation de dimension 60 on va se poser les questions suivantes :

— Est-ce que tous ces paramètres ont la même importance ?

— Si non, quels sont les plus importants ?

Prenons dans ce cas :

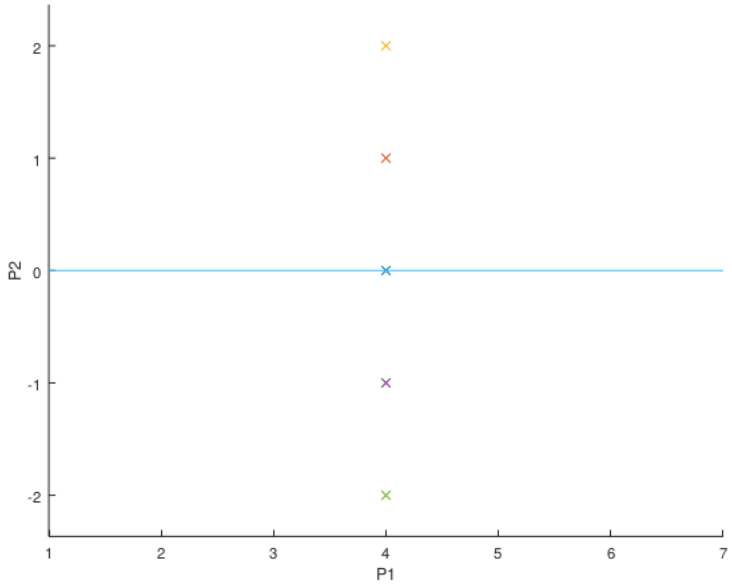


FIGURE 5.3 – Exemple de paramètres inutiles

Dans cet exemple il est clair que les paramètres P1 sont inutiles pour établir un choix : **P1 est identique pour tous les échantillons**.

Certaines méthodes ou algorithmes permettent une très grande réduction. Si nous plaçons dans $\mathbb{R}^{60}$ avec 300 échantillions un méthode de réduction serait par exemple d'utiliser le **DeepLearning**. Cette méthode va permettre de réduire jusqu'à $\mathbb{R}^3$ en offrant un bon processus de sélection [1].

5.3 Méthodes de réduction

5.3.1 Méthode ACP(Analyse en composante principale)

Il existe deux méthodes pour réaliser une réduction en dimension. La premère est **l'Analyse en composante principale**. Le but est réduire l'observation de manière globale en essayant de conserver le maximum d'information.
On oublierait la classe et on essaie de réduire en conservant au mieux la **variance**.

1. tout en risquant une perte d'informations.

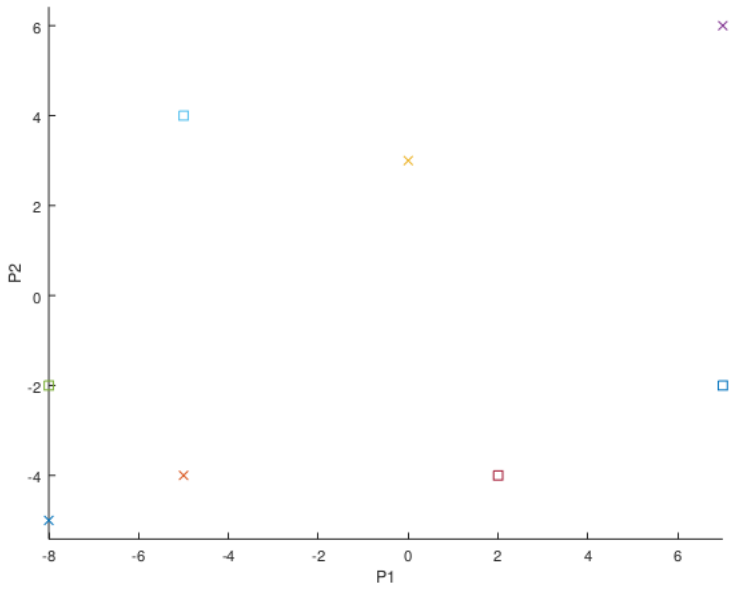


FIGURE 5.4 – Avant l'ACP

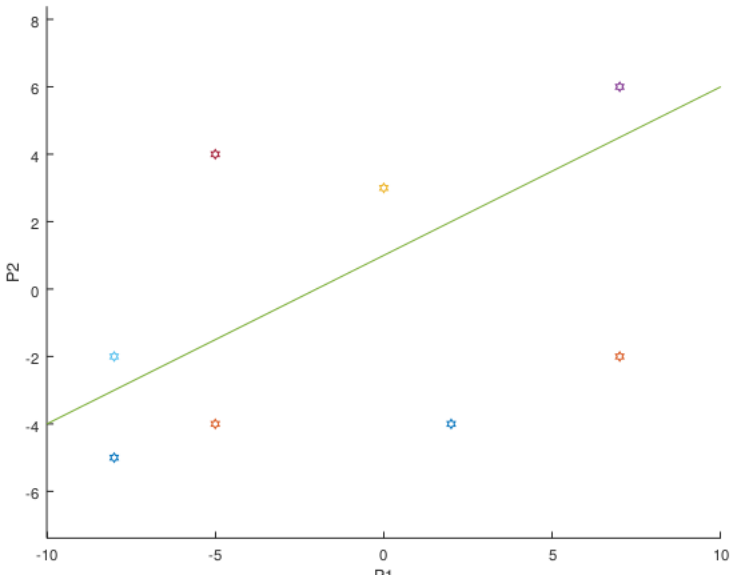


FIGURE 5.5 – Après l'ACP

5.3.2 Méthode ACI(Analyse en composantes Indépendantes)

Cette fois, la réduction va se faire en tenant compte de l'observation et en essayant de trouver un projection qui sépare au mieux les classes.

En bref, on conserve les classes et on cherche une projections des données qui les sépare au mieux.

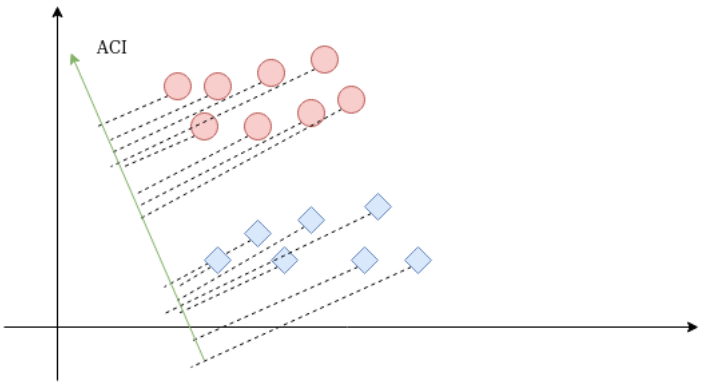


FIGURE 5.6 – Exemple d'ACI

À savoir : la méthode ACI permet une réduction de dimension entre 1 et le **le nombre de classe** -1. Ainsi il est possible de passer de $\mathbb{R}^{60}$ à $\mathbb{R}$

5.3.3 ACP : fonctionnments et calculs

Dans cette section on verra comment on passe de $\overrightarrow{p} = \{p_1, ..., p_d\}$, le vecteur de l'observation à $\overrightarrow{y} = \{y_1, ..., y_k\}$ un vecteur de points projetés.

Recentrage des données

On va recentrer les points de l'observation en passant de $\overrightarrow{p}$ à $\overrightarrow{c}$ tel que :

$$c_i = p_i - \mu$$

Avec :

$$\mu = \frac{1}{N} \sum_{i=1}^{N} p_i, N = d$$

Travail à réaliser

Le but de la project peut être formalisé ainsi :

$$\overrightarrow{y} = W^T \overrightarrow{c}$$

$$\begin{bmatrix} y_1 \\ \dots \\ y_k \end{bmatrix} = \begin{bmatrix} e_{1,1} & \dots & e_{1,d} \\ \dots & \dots & \dots \\ e_{k,1} & \dots & e_{k,d} \end{bmatrix} \begin{bmatrix} c_1 \\ \dots \\ c_d \end{bmatrix}$$

On notera que l'on va passer d'une dimensions $\mathbb{R}^d$ à une dimensions $\mathbb{R}^k$.

Base

Les colonnes de la matrice W,sont de la forme [2] :

$$\overrightarrow{e}_i = \begin{pmatrix} e_{1,i} \\ e_{2,i} \\ \dots \\ e_{d,i} \end{pmatrix}$$

Ces vecteurs forment une base, du sous-espace vectoriel de dimension k. On considèrera que la base est orthonormée, ce qui donne les propriétées suivantes :

— $\forall i \neq j, \overrightarrow{e}_i . \overrightarrow{e}_j = 0$
— $\forall i, \overrightarrow{e}_i . \overrightarrow{e}_i = 1$

Fonction à minimiser

On donne la fonction suivante :

$$J(\overrightarrow{e}_1 \dots \overrightarrow{e}_k, \alpha_{1,1} \dots \alpha_{n,k}) = \sum_{i=1}^{N} ||c_i - \tilde{c_i}||_2$$

avec :

$\tilde{c_i} = \sum_{j=1}^{k} \alpha_{i,j} \overrightarrow{e}_j$: le projeté de c_i dans le sous-espace défini par la base

$||c_i - \tilde{c_i}||_2$: l'erreur commise sur un échantillon

On peut finalement donner la définition de la fonction précédente :

$$J(\overrightarrow{e}_1 \dots \overrightarrow{e}_k, \alpha_{1,1} \dots \alpha_{n,k}) = \sum_{i=1}^{N} ||c_i - \sum_{j=1}^{k} \alpha_{i,j} \overrightarrow{e}_j||_2$$

Dérivé de J sur α

On va partiellement dériver la fonction J sur $\alpha_{i,j}$ afin d'en connaître la valeur :

$$\frac{\partial J(\overrightarrow{e}_1 \dots \overrightarrow{e}_k, \alpha_{1,1} \dots \alpha_{n,k})}{\partial \alpha_{i,j}} = 0$$

On passera les calculs pour obtenir la formule suivante :

$$\alpha_{i,j} = c_i^T \overrightarrow{e}_j$$

En faisant abstraction de calculs supplémentaires on arrive à devoir résoudre le système suivant :

$$S\overrightarrow{e}_i = \lambda_i \overrightarrow{e}_i$$

2. Attention ! On parle bien de W et non pas W^T.

avec :

$$S = \sum_{i=1}^{N} c_i c_i^T = \sum_{i=1}^{N} (p_i - \mu)(p_i - \mu)^T : \text{S est appelée Scatter Matrix}$$

Les solutions de ce système permettront de trouver une minimisation de la fonction $J(\overrightarrow{e}_1 \dots \overrightarrow{e}_k)$. Ces solutions sont les **vecteurs propres** de la matrice S. On remarquera que S est de rang d, c'est à dire qu'elle possède d **valeurs propres**.

On formera des couples $(\overrightarrow{e}_i, \lambda_i)$ que l'on classera en fonction de la valeur de λ_i

Exemple avec une base $B = (\vec{e}_1)$

On va ici les c_i sur $\mathbb{R}$:

$$\text{Perte d'info} : \frac{1 - \lambda_1}{\sum_{i=1}^{d} \lambda_i}$$

$$\text{Info préservée} : \frac{\lambda_1}{\sum_{i=1}^{d} \lambda_i}$$

Exemple avec une base $B = (\vec{e}_1, \vec{e}_2)$

On va ici les c_i sur $\mathbb{R}^2$:

$$\text{Perte d'info} : \frac{1 - \lambda_1 + \lambda_2}{\sum_{i=1}^{d} \lambda_i}$$

$$\text{Info préservée} : \frac{\lambda_1 + \lambda_2}{\sum_{i=1}^{d} \lambda_i}$$

Exemple avec une base $B = (\vec{e}_1, ..., \vec{e}_d)$

On va ici les c_i sur $\mathbb{R}^d$:

$$\text{Perte d'info} : 0$$

$$\text{Info préservée} : 1$$

Marche à suivre résumée

Pour $P = \{p_1, ..., p_N\}$ un ensemble de points :

1. calculer $\mu = \frac{1}{N} \sum_{j=1}^{N} p_j$
2. recentrer P en $C = \{c_1, ..., c_N\}, c_i = (p_i - \mu)$
3. calculer $S = \sum_{j=1}^{N} c_j c_j^T$
4. calculer les couples de S $(\lambda_i, \overrightarrow{e}_i)$

5. retenir les k vecteurs propres $(\overrightarrow{e}_1, ..., \overrightarrow{e}_k)$ des plus grandes valeurs propres, avec $(\lambda_1 \leq \cdots \leq \lambda_k)$. On trie les vecteurs propres, en fonctions des valeurs des valeurs propres
6. former $W^T = \begin{bmatrix} e_{1,1} & \dots & e_{1,d} \\ \dots & \dots & \dots \\ e_{k,1} & \dots & e_{k,d} \end{bmatrix}$
7. faire la projection des c_i

Chapitre 6

Analyse discriminante

Cette fois ci, on va chercher la projection qui sépare *au mieux* les classes. On garde donc l'information de la classe dans le processus de réduction.

6.1 Calculs

6.1.1 Moyenne des projetés

Pour ce faire on choisira comme critères "*au mieux*" de faire attention à la moyenne de la projection des classes [1] :

$$(\tilde{\mu}_1 - \tilde{\mu}_2)^2$$

Cependant, il faut en plus, s'assurer de minimiser la somme des variances, notées : $\hat{S}_1^2$ et $\hat{S}_2^2$ tel que :

$$\frac{1}{\hat{S}_1^2 + \hat{S}_2^2}$$

Ainsi on obtient la fonction à minimiser suivante :

$$J(W) = \frac{(\tilde{\mu_1} - \tilde{\mu_2})^2}{\hat{S}_1^2 + \hat{S}_2^2}$$

W est la matrice de projection. On donne les définition des $\tilde{\mu_i}$:

$$\tilde{\mu_i} = \frac{\sum_{j=1}^{|C_i|} W^T c_j}{|C_i|}$$

$$= \frac{1}{|C_i|} W^T \sum_{j=1}^{|C_i|} c_j$$

1. on travaillera ici avec 2 classes.

$$= W^T \frac{1}{|C_i|} \sum_{j=1}^{|C_i|} c_j$$

$$\tilde{\mu_i} = W^T \mu_i$$

On peut maintenant travailler sur la formule suivante :

$$(\tilde{\mu_1} - \tilde{\mu_2})^2 = W^T(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T W$$

$$\text{simplifiable par, } W^T S_b W, \quad S_b = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

$$(\tilde{\mu_1} - \tilde{\mu_2})^2 = W^T S_b W$$

6.1.2 Variance

S_k avec $k \in \{1, 2\}$.

$$S_k = \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} (c_i - \mu_k)(c_i - \mu_k)^T$$

On donne donc :

$$\hat{S}_k^2 = \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} (\tilde{y_i} - \tilde{\mu_k})^2$$

$$= \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} (W^T c_i - \tilde{\mu_k})^2$$

$$= \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} (W^T c_i - \tilde{\mu_k})^2$$

$$\hat{S}_k^2 = \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} W^T (c_i - \tilde{\mu_k})^T (c_i - \tilde{\mu_k}) W$$

$$\hat{S}_k^2 = \frac{1}{|C_k|} \sum_{i=1}^{|C_k|} W^T S_K W$$

6.1.3 Nouvelle forme de la fonction à minimiser

$$J(W) = \frac{W^T S_b W}{W^T S_w W}, S_w = \sum S_k$$

On dérive partiellement $J(W)$ sur W :

$$\frac{\partial J(W)}{\partial W} = 0$$

$$S_b W - \alpha S_w W = 0$$
$$S_b W = \alpha S_w W$$
$$S_w^{-1} S_b W = \alpha W$$
$$S_w^{-1} S_b W = S_w^{-1}(\alpha(\mu_1 - \mu_2))$$

On sait grâce à cela que :

$$W = S_w^{-1}(\mu_1 - \mu_2)^T$$

[2]

6.2 Généralisation à n classes

$$S_b = \sum_{i=1}^{n} (\tilde{\mu_i} - \mu)(\tilde{\mu_i} - \mu)^T, \mu = \frac{\sum_{i=1}^{N} \tilde{\mu_i}}{N}$$

$$S_w = \sum_{i=1}^{n} \hat{S_i}$$

$$\hat{S_k} = \frac{1}{|C_k|} \sum_{i=1}^{k} (c_i - \mu_k)(c_i - \mu_k)^T$$

Ces formules nous amènent à trouver :

$$S_b W = \lambda S_w W$$

$$S_w^{-1} S_b W = \lambda W$$

On précise que dans ce cas $S_w^{-1} S_b$ es tde rang $N - 1$

6.3 Que retenir ?

- Calculer S_b
- Calculer S_w
- $J(W) = \frac{W^T S_b W}{W^T S_w W} = \frac{(\tilde{\mu_1} - \tilde{\mu_2})^2}{\hat{S_1^2} + \hat{S_2^2}}$, dans le cas à deux classes.
- Dans le cas N classe on peut réduire de 1 à N-1

2. ajout de la transposé en TD

Chapitre 7

Classifieurs linéaires

On utilise un classifieur linéaire lorsque l'on travaille sur une vérité terrain à n classes. On va chercher à trouver un ensemble d'**hyperplan** qui sépare les classes.

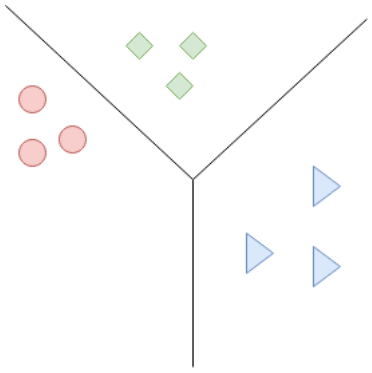


FIGURE 7.1 – Exemple de cas pouvant nécessiter un classifieur linéaire

7.1 Utilisation du problème à 2 classes

7.1.1 Explication du problème

On verra que l'on peut se servir du problème à deux classe pour le problème à n classes, cependant il existe une **marge** d'erreur.

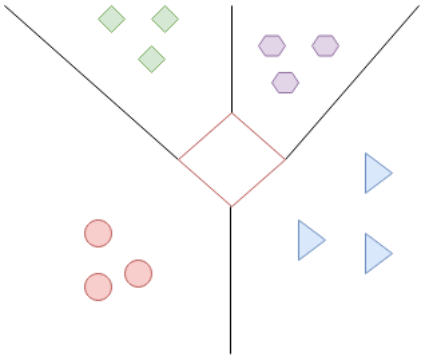


FIGURE 7.2 – Exemple de risques,lié à l'utilisation du problème à 2 classess pour n classes

7.2 Classifieur linéaire à deux classes

Dans ce classifieur linéaire on peut obtenir une fonction $g(x)$:

$$g(x) = W^T x + W_0$$

W^T est un vecteur normal à l'hyperplan, et W_0 un décalage à l'origine. On défini donc un hyperplan par g qui sépare l'espace en 2 palier :

$$g(x) > 0 \rightarrow x \in C_1$$

$$g(x) < 0 \rightarrow x \in C_2$$

$$g(x) = 0, \textbf{On se sait pas}$$

Un problème ne sera linéairement séparable seulement si il existe (W, W_0) tel que :

- $\forall x_i \in C_1 : g(x_i) = W^T x_i + W_0 > 0$
- $\forall x_i \in C_2 : g(x_i) = W^T x_i + W_0 < 0$

7.2.1 Premièr transformation de g

On va maintenant transformer $g(x)$ en $\tilde{g}(\tilde{x})$:

$$\tilde{g}(\tilde{x}) = \tilde{W}\tilde{x}, \tilde{W} = \begin{bmatrix} W_0 \\ W \end{bmatrix}$$

et

$$\tilde{x} = \begin{bmatrix} 1 \\ x \end{bmatrix}$$

Il faut donc aussi réaliser une extension de la vérité terrain :$VT \rightarrow \tilde{VT}$. On réalise cela en rajoutant 1 devant tous les élémentsde l'observation :
Exemple : $(a, b) \rightarrow (1, a, b)$.

7.2.2 Seconde transformation de g

On transforme $\tilde{g}(\tilde{x})$ en $\tilde{\tilde{g}}(\tilde{\tilde{x}})$, tel que $\forall \tilde{\tilde{x}} \in \tilde{\tilde{C}}_1 \cup \tilde{\tilde{C}}_2$.
Si $\tilde{g}(\tilde{x}) = \tilde{\tilde{g}}(\tilde{\tilde{x}}), \forall \tilde{x} \in \tilde{C}_1$, alors $\tilde{C}_1 = \tilde{\tilde{C}}_1$.
Comment faire pour $\tilde{C}_2$?
Pour ce cas si on va transformer $\tilde{\tilde{x}} \in \tilde{\tilde{C}}_2$ tel que $-\tilde{g}(x) = \tilde{\tilde{g}}(\tilde{\tilde{x}})$
Ainsi on obtient :$g(-\tilde{x}) = \tilde{\tilde{g}}(\tilde{\tilde{x}})$ et $\tilde{\tilde{C}}_2 = (-1)\tilde{C}_2$ On vient de faire les transformations suivantes :

$$VT \rightarrow \tilde{VT} \rightarrow \tilde{\tilde{VT}}$$

7.2.3 Changement de notation

$$\tilde{\tilde{W}} = a$$
$$\tilde{\tilde{W}}^T = a^T$$

et

$$\tilde{\tilde{x}} = y$$

Et donc on redéfini le problème en cherchant :

$$\forall y \in \text{Observation tranformé} = \{\tilde{\tilde{C}}_1 \cup \tilde{\tilde{C}}_2\}$$
$$a^T y \geq 0$$

7.2.4 Que fait-on dans le cas où $a^T < 0$?

Si $a^T y < 0$ alors on peut dire que a n'est pas solution du problème et que y est mal classé par a. On définira alors :

$$Y_m(a) = \{y \in \tilde{\tilde{\text{obs}}}\}, a^T y < 0$$

Il s'agit là de l'ensemble des y tel que $a^T y < 0$.
Si a est solution alors $Y_m(a) = \emptyset$. On doit trouver une fonction $J(a)$ dont a est la solution du problème [1] quand $J(a)$ est minimal.

On donne alors $J(a) = |Y - m(a)|$, le nombre de mal classés. Si $|Y - m(a)| = 0$ alors a est solution.
Le problème de cette fonction est qu'elle est constante par morceau :

1. séparateur linéaire

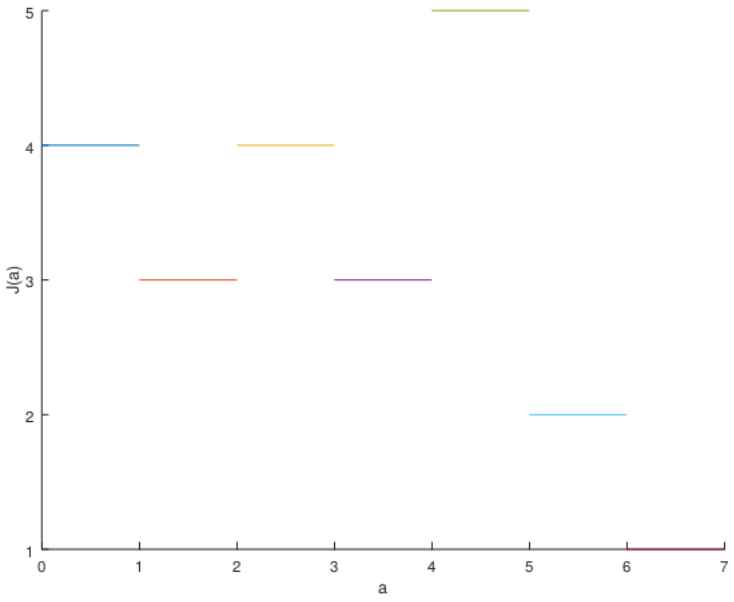


FIGURE 7.3 – J(a) est constante par morceau

Comme cette fonction est constante, on ne peut pas faire de descente de gradient.
On redéfinie alors la fonction $J(a)$:

$$J(a) = \sum_{y \in Y_m(a)} \frac{a^T y}{||a||}$$

On obtient un problème de minimisation que l'on va transformer en problème de maximisation.

$$J(a) = -\sum_{y \in Y_m(a)} \frac{a^T y}{||a||}$$

Ainsi si $Y_m(a) = \emptyset \leftarrow J(a) = 0$ et cette fonction est cette fois ci continue par morceaux :

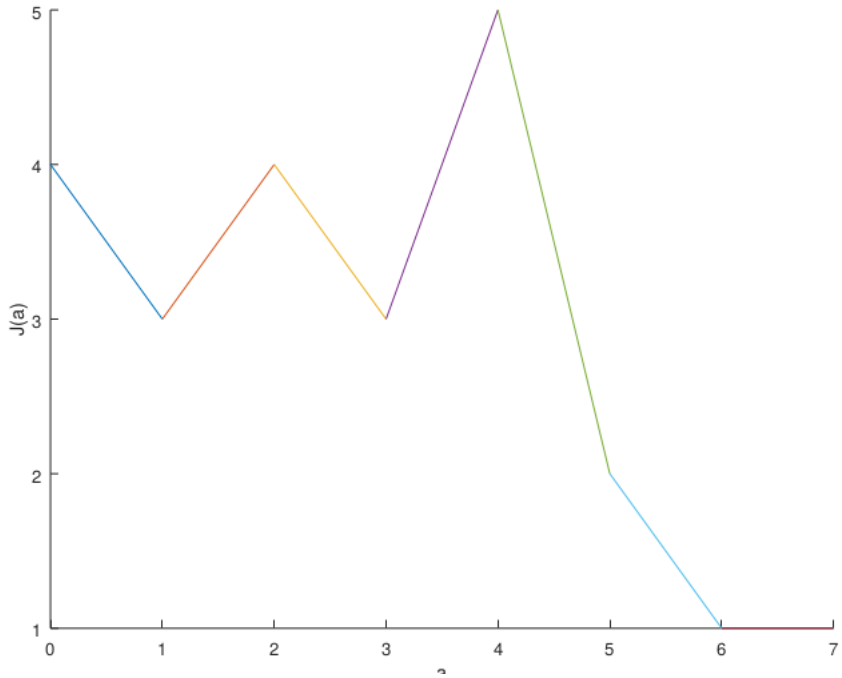


FIGURE 7.4 – J(a) est continue par morceaux

On peut remarquer que l'on peut simplifier la fonction à maximiser :

$$J(a) = -\sum_{y \in Y_m(a)} \frac{a^T y}{||a||} \iff J(a) = -\sum_{y \in Y_m(a)} a^T y$$

$$\iff J(a) = \sum_{y \in Y_m(a)} -a^T y$$

Dérivée

Pour maximiser il nous faut la fonction dérivée de $J(a)$:

$$\frac{\partial J(a)}{\partial a} = -\sum_{y \in Y_m(a)} y$$

Le problème que l'on a désormais vient du fait que l'on a pas d'expression analytique pour $Y_m(a)$. Tel que, pour calculer $Y_m(a)$ il faut passer en revue, l'ensemble des éléments de l'observation.

On utilise donc un descente de gradient :

$$a_0, \text{à calculer ou choisir}$$

$$a_k = a_0, \text{à l'initialisation}$$

$$a_{k+1} = a_k + \sum_{y \in Y_m(a)} y$$

Exemple :

On prend comme valeurs :

$$\tilde{\tilde{obs}} = \{(1,1,2),(1,3,2),(-1,-1,-1),(-1,-2,-3)\}$$

$$a_0 = [1,-2,2]$$

On peut donc commencer les calculs :

$$Y_m(a_0) = ?$$

$$a_0^T y_1 = [1,-2,2] \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = 3 \text{ bien classé}$$

$$a_0^T y_2 = [1,-2,2] \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 3 \text{ bien classé}$$

$$a_0^T y_3 = [1,-2,2] \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} = -1 \text{ mal classé}$$

$$a_0^T y_4 = [1,-2,2] \begin{bmatrix} -1 \\ -3 \\ -2 \end{bmatrix} = 1 \text{ bien classé}$$

$$Y_m(a_0) = \{y_3\} = \{(-1,-1,-1)\}$$

Ainsi on calcul a_1 :

$$a_1 = a_0 + \sum_{y \in Y_m(a0)} y = \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix} + \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ -3 \\ 1 \end{bmatrix}$$

Puis on réitère et on obtient :

$$a = [0,-4,3]$$

Conclusion

Ainsi pour évaluer a_k on possède deux stratégies :

1. Calculer toute l'observation, $Y_m(a_k)$
2. On prend le premier mal classé et on l'ajoute à a_k

Ainsi :

— Si le problème est linéairement séparable, l'algorithme converge vers un des solutions.

— Si le problème n'est pas linéairement séparable, alors il n'existe pas de a_k tel que $y_m(a_k) = \emptyset$ on entre dans une boucle infinie.

Pour résoudre la deuxième situation on va atténuer la contribution du gradient via un coefficient $\eta(k)$:

$$a_{k+1} = a_k + \eta(k) \sum_{y \in Y_m(a0)} y$$

Ainsi dans le cas où le problème n'est pas linéairement séparable, la solution dépendra de a_0 et aussi de $\eta(k)$.

7.2.5 Système d'équation

On a cherché a tel que $\forall y \in \tilde{\tilde{obs}}, a^T y > 0$.

On donne une nouvelle approche :

Soit $b \in \mathbb{R}^{n+1}$, on cherche a tel que $Ya = b$ avec :

$$Y = \begin{bmatrix} e_i \in \tilde{\tilde{C}}_1 \\ e_i \in \tilde{\tilde{C}}_2 \\ \dots \\ e_i \in \tilde{\tilde{C}}_j \end{bmatrix} \quad \text{la matrice des éléments de l'observation}$$

$$b = \begin{bmatrix} b_1 > 0 \dots \\ b_{max} > 0 \end{bmatrix}$$

avec $max = |\tilde{\tilde{Obs}}|$.

On donne ainsi :

$$Ya = \begin{bmatrix} (Y_1 a^T) = (a^T Y_1) \\ \dots \\ (Y_k a^T) = (a^T Y_k) \end{bmatrix}$$

$$Ya = b \rightarrow \begin{bmatrix} (Y_1 a^T) = (a^T Y_1) \\ \dots \\ (Y_k a^T) = (a^T Y_k) \end{bmatrix} = \begin{bmatrix} b_1 > 0 \dots \\ b_{max} > 0 \end{bmatrix}$$

Remarque

$a^T > 0$ englobe $Ya = b$ car on fixe b dans $Ya = b$ ce qui revient à dire que $Ya = b$ est un sous problème de $a^T y > 0$

7.2.6 Avec le vecteur b

Il y a une solution si le nombre d'éléments de l'observation est égal à la dimension de l'espace qui définit a.

Ainsi, Si $x \in obs, x \in \mathbb{R}^2$ et $y \in \tilde{\tilde{obs}}, y \in \mathbb{R}^3$. Il faut 3 éléments dans l'observation.
Si on a pour a, 3 inconnues alors $a \in \mathbb{R}^3$. ainsi si on a 200 observations, on obtient un système à 200 équations et à 3 inconnues. On peut facilement affirmer que le système n'as pas une solution unique et que :

$$Ya \approx b$$

C'est donc la fonction suivante que l'on va minimiser :

$$J(a) = ||Ya - b||^2, \text{ avec b choisi}$$

Comment obtenir cette fonction ?

On chercher a tel que $\forall y_i \in \tilde{\tilde{Obs}}, a^T y_i > 0$
À partir de ceci on souhaite calculer $Ya - b$. Y est un vecteur où chaque y_i correspond à la distance entre un élément de l'observation et la projection pour a.
On associe donc à un y_i un a

$$y_i, a : a^T y_i - b_i$$

On donne ainsi :

$$\begin{aligned} J(a) &= \sum_{i=i}^{m} (a^T y_i - b_i)^2 \\ &= ||Ya - b||^2 \end{aligned}$$

Dérivée

La dérivé partielle de cette fonction est la suivante :

$$\frac{\partial J(a)}{\partial a} = 2y^T(ya - b)$$

Donc, si $2y^T(ya - b) = 0 \implies \nabla J(a) = 0$.

À partir de là il y a deux solutions :

1. résolution de $2y^T(ya - b)$, soit une résolution (approximée) du système d'équation.
2. un descente de gradient

Résolution du système

On souhaite résoudre :

$$2y^T(ya - b) = 0$$

Que l'on transforme en :

$$Y^T(Ya - b = 0)$$

$$Y^TYa = Y^Tb$$

On notera que Y^TY est une matrice carrée.
Si Y^TY est inversible alors $(Y^TY)^{-1}$ existe. Ainsi on dit :

$$(Y^TY)^{-1}Y^TYa = (Y^TY)^{-1}Y^Tb$$

$$a = (Y^TY)^{-1}Y^Tb$$

On obtient alors ici un a qui fait de $Ya \approx b$ une bonne approximation. Au final on obtient la relation suivante :

$$Ya \approx b \approx \begin{bmatrix} b_1 + \epsilon_1 \\ \dots \\ b_n + \epsilon_n \end{bmatrix}$$

Cependant rien ne garantie que $\epsilon_i = 0$. De ce fait on ne peut pas garantir que la a obtenue sépare linéairement le problème et ce même si le problème est linéairement séparable

On se rend compte que la solution va dépendre de la valeur de b.

Si $b = \begin{bmatrix} 1 \\ \dots \\ 1 \end{bmatrix}$, alors $||Ya - \begin{bmatrix} 1 \\ \dots \\ 1 \end{bmatrix}||^2 \to a$.
Et si Si $b = \begin{bmatrix} 2 \\ \dots \\ 2 \end{bmatrix}$, alors $||Ya' - \begin{bmatrix} 2 \\ \dots \\ 2 \end{bmatrix}||^2 \to a'$ alors :

$$a' = 2a$$

$$||Y2a - \begin{bmatrix} 2 \\ \dots \\ 2 \end{bmatrix}||^2 = 2||Ya - \begin{bmatrix} 1 \\ \dots \\ 1 \end{bmatrix}||^2$$

On voit bien que b n'intervient pas dans la détermination de a
Si $\forall i, b_i =$ constante alors la solution peut se ramener à $\begin{bmatrix} 1 \\ \dots \\ 1 \end{bmatrix}$.
Si a est solution pour $b = \begin{bmatrix} 1 \\ \dots \\ 1 \end{bmatrix}$ alors constante $*a$ est solution de $b = \begin{bmatrix} \text{constante} \\ \dots \\ \text{constante} \end{bmatrix}$.

Ainsi on pose :

$$Ya = \begin{bmatrix} a^T y_1 \\ \dots \\ a^T y_n \end{bmatrix} \text{ avec } a^T y_i < 0$$

Et de ce fait :

$$Y\text{constante } a \begin{bmatrix} \text{constante} a^T y_1 \\ \dots \\ \text{constante} a^T y_n \end{bmatrix} \text{ et constante } a^T y_i < 0$$

Cas à prendre en compte : si $b = \begin{bmatrix} 1 \\ \dots \\ 1 \\ 10 \end{bmatrix}$ On voit que le dernier terme de ce vecteur est différent : il représente un traitement particulier. Notamment, le point associé sera plus loin que les autres dans la solution.

La valeur de ce b_i pondère la distance enter la projection du i-ème échantillon et la droite obtenue

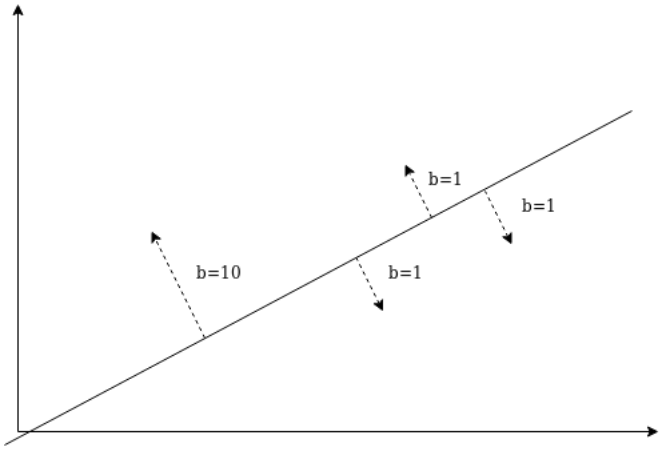


FIGURE 7.5 – Plus b est grand plus la distance est grande

Comme on le voit **plus b_i est grand, plus la distance est grande**, ce qui veut aussi dire que **moins b_i contribue par rapport aux autres**

Descente de gradient en fonction de a

On donne l'algorithme de descente de gradient suivant :

$$\text{Tant que } |a_{k+1}| \geq \epsilon$$

$$a_{k+1} = a_k - \eta(k)\nabla J(a_k)$$

On rappelle que $\nabla J(a_k) = 2Y^T(Ya - b)$, cependant il s'agit là de données difficiles à stocker. Ainsi on se propose de le faire élément par élément :

$$a_{k+1} = a_k - \eta(k)2y_i[(a_k^T y_i) - b_i]$$

Cette descente de gradient se fait uniquement en fonction de a,b lui étant fixé. Si jamais après la descente, a n'est pas solution Ya, il faut changer la valeur de b

Descente de gradient en fonction de a et de b

On fait maintenant minimiser en fonction de a et de b :

$$J(a, b) = ||Ya - b||^2, b \geq 0$$

On se donne un a et on approxime b. On commence par calculer une dérivée partielle :

$$\frac{\partial J(a, b)}{\partial b} = 2(Ya - b)$$

Si on essaies de résoudre :

$$2(Ya - b) = 0$$

$$Ya = b$$

Il n'y a pas de solution analytique.

On effectue donc une descente de gradient :

$$b_{k+1} = b_k - 2(Ya - b)$$

$$= b_k - \eta(k)2(Ya - b)$$

Cependant avec cette descente de gradient il se peut que b_{k+1} prenne une valeur négative. On donne alors le nouvel algorithm suivant avec :

$$e_k = Ya - b_k$$

$$b_{k+1} = b_k + \eta(e_k + |e_k|)$$

Si $e_k > 0 \rightarrow e_k + |e_k| \implies 2e_k$. Sinon $e_k < 0 \rightarrow e_k + |e_k| \implies 0$.

Algorithme Final :

1. On se donne $b_0 > 0$
2. On calcul a_k en utilisant :
   — la pseudo-inverse
   — la descente de gradient
3. On bloque a_k obtenu
4. On fait une descente gradient en utilisant a_k et on calcule b_k (on remonte à 2 si besoin)

On s'arrêtera quand :

$$b_{k+1=b_k+\epsilon}$$

ou alors quand :

$$a_{k+1} = a_k + \epsilon$$

Problèmes

Que l'on utilise l'algorithme du Perceptron ou que l'on calcul une approximation :

On ne marise pas la position de la droite/hyperplan

La droite doit être aussi près de certains éléments de C_1 que de C_2 :

$$min(dist(D, y \in C_1)) = min(dist(D, y \in C_2))$$

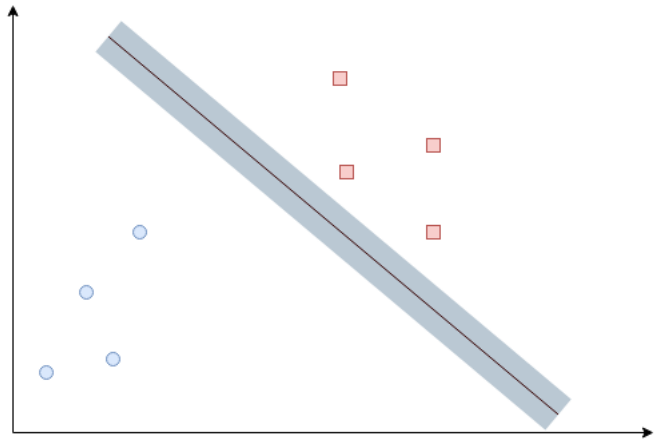


FIGURE 7.6 – la droite doit assez proches des différents éléments des classes