Machine Learning

Dr KEITA Kolé

Université Musulmane Africaine
UFR Sciences Economiques et de Gestion

Année Universitaire 2023-2024

Sommaire

Exemple 1 :

La base de données ci-dessous porte sur les mouvements journaliers d'indices boursiers de Standard & Poor's 500 (500 grandes sociétés cotées sur les bourses aux États-Unis) sur 5 ans. Source : https://finance.yahoo.com/

Year	Lag1	Lag2	Lag3	Lag4	Lag5	Volume	Today	Direction
2001	0.381	−0.192	−2.624	−1.055	5.010	1.1913	0.959	Up
2001	0.959	0.381	−0.192	−2.624	−1.055	1.2965	1.032	Up
2001	1.032	0.959	0.381	−0.192	−2.624	1.4112	−0.623	Down
2001	−0.623	1.032	0.959	0.381	−0.192	1.2760	0.614	Up
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2005	0.043	0.422	0.252	−0.024	−0.584	1.28581	−0.955	Down
2005	−0.955	0.043	0.422	0.252	−0.024	1.54047	0.130	Up
2005	0.130	−0.955	0.043	0.422	0.252	1.42236	−0.298	Down
2005	−0.298	0.130	−0.955	0.043	0.422	1.38254	−0.489	Down

Avec

- Lag1 : le pourcentage de la variation pour le jour précédent
- Lag1 : le pourcentage de la variation pour le jour d'après
- ...
- Volume : le nombre d'actions négociées quotidiennement
- Today : le pourcentage de rendement
- Direction : une variable binaire qui indique si le marche est négatif ou positif

Objectif : prédire la variable catégorielle Direction qui indique la performance du marché en fonction des pourcentages de variable des indices journaliers.

Il s'agit d'un problème de **classification**.

Exemple 2 :

Techniques de statistique de filtrage automatique des spams : le volume croissant de courriers électroniques non sollicités (appelés « spam ») a généré un besoin de filtres anti-spam fiables.
La base de données utilisée dans [1] contient 4601 messages électroniques.

Objectif : concevoir un détecteur automatique capable de filtrer les messages électroniques avant d'encombrer les boîtes mail des utilisateurs. Il s'agit de prédire si un mail est spam ou non.

Pour l'ensemble des 4601 messages, le véritable résultat est disponible, ainsi que les fréquences relatives de 57 mots et signes de ponctuation les plus courants dans le message électronique.

[1] Source : Mark Hopkins, Erik Reeber, George Forman, Jaap Suermondt, Hewlett-Packard Labs, USA :
https : //mlr3.mlr − org.com/reference/mlr$_t$asks$_s$pam.html

Le tableau ci-dessous donne des mots et des caractères affichant la plus grande moyenne dans le spam et le courrier électronique.

	george	you	your	hp	free	hpl	!	our	re	edu	remove
spam	0.00	2.26	1.38	0.02	0.52	0.01	0.51	0.51	0.13	0.01	0.28
email	1.27	1.27	0.44	0.90	0.07	0.43	0.11	0.18	0.42	0.29	0.01

Il s'agit d'un problème de **classification** dont les classes de la variable catégorielle (réponse) sont message et spam.

Exemple 3 : Reconnaissance de chiffres manuscrits

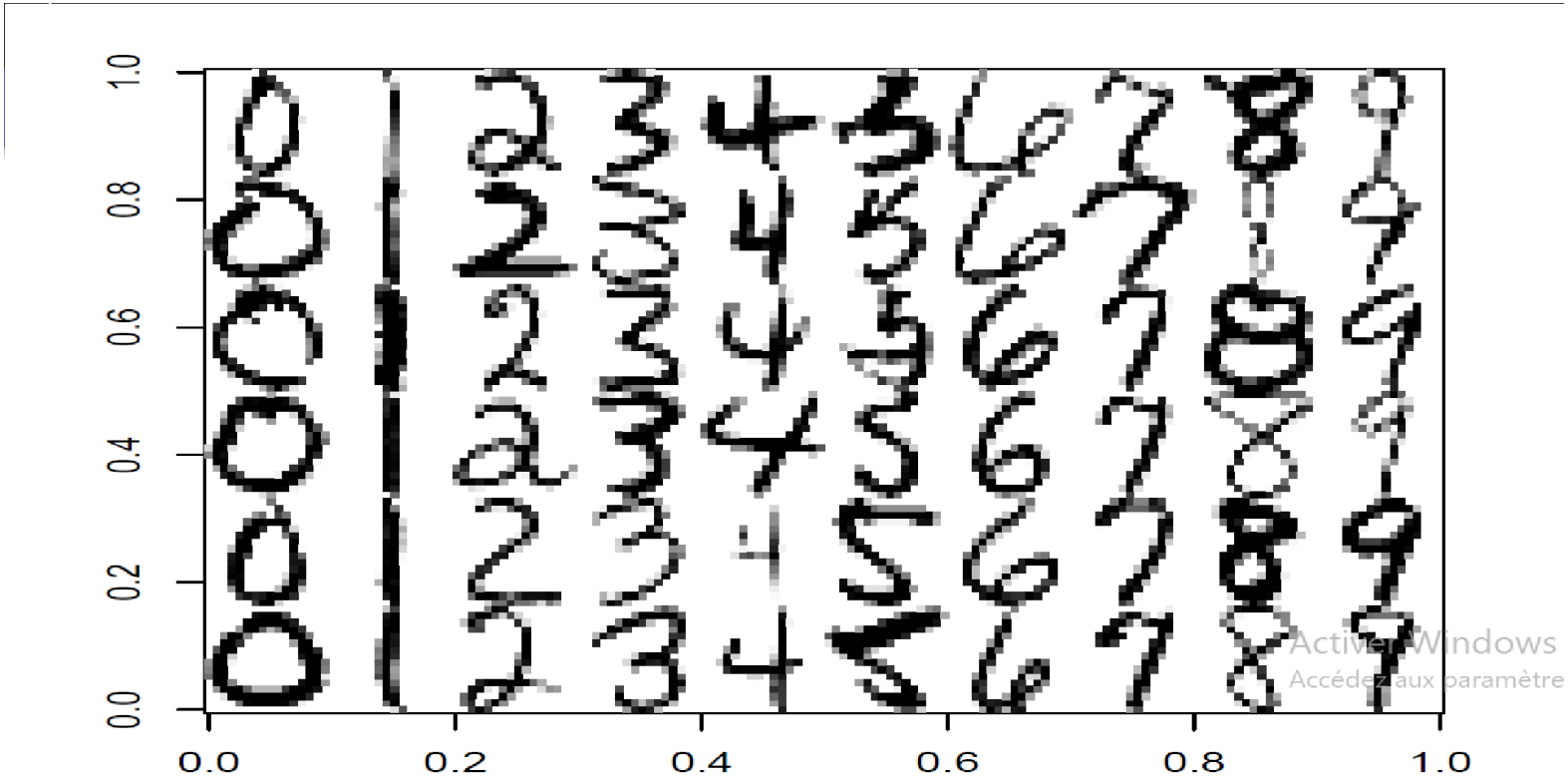


Figure: Exemples de chiffres manuscrits provenant d'enveloppes postales américaines [1]

[1] Source : AT&T Bell Labs, USA

[1] Source : AT&T Bell Labs, USA

L' image correspond aux données provenant des codes postaux manuscrits scannés à partir d'enveloppes du service postal américain. Chaque caractère est un seul chiffre, isolé d'un code postal à 5 chiffres.
Les caractères sont des images grises de 16×16 bits, chacune pixel dont l'intensité varie de 0 à 255. Les caractères ont été normalisés pour avoir approximativement la même taille et la même orientation.

Objectif : prédire l'identité d'une nouvelle image $c \in \{0, 1, 2, \cdots, 9\}$ de 16 x 16 pixels.

Il s'agit encore d'un problème de **classification**.

Exemple 4 : Scoring

- Le scoring est un domaine de la statistique décisionnelle dont le but est de discriminer, de sélectionner, de classer, de segmenter, de prévoir le comportement d'un client conformément à un critère donné. Il existe plusieurs types de scores au sein des banques :
  - **acceptation d'une offre de prêt**,
  - **fraude** (transaction,...),
  - **retard de paiement**, etc.
- Ces techniques sont aussi appliquées en marketing : optimiser ses actions commerciales en envoyant des offres à des clients sélectionnés. Plus l'entreprise connaitra ses clients, plus elle sera susceptible de leur proposer des produits personnalisés.
  D'autres études de scoring orientées Marketing :
  - scores d'**appétence** : évaluer les probabilités qu'un client réponde favorablement à une offre ou à un service proposé.
  - scores d'**attrition** : traduire la probabilité qu'un client ou un abonné passe chez les concurrents ou résilie son abonnement.

- Score d'assurance évalue la probabilité qu'un client soit impliqué dans un futur accident ou une réclamation d'assurance (un score favorable entraînera une baisse de paiement).

Le score a pour but de classer les emprunteurs pour prédire la classe D (**défaut**) et la classe ND (**non défaut**) dans laquelle nous allons ensuite les observer.

Les revenus annuels et les soldes mensuels de cartes de crédit pour un sous-ensemble de 10000 personnes sont représentés sur la figure ci-dessous.

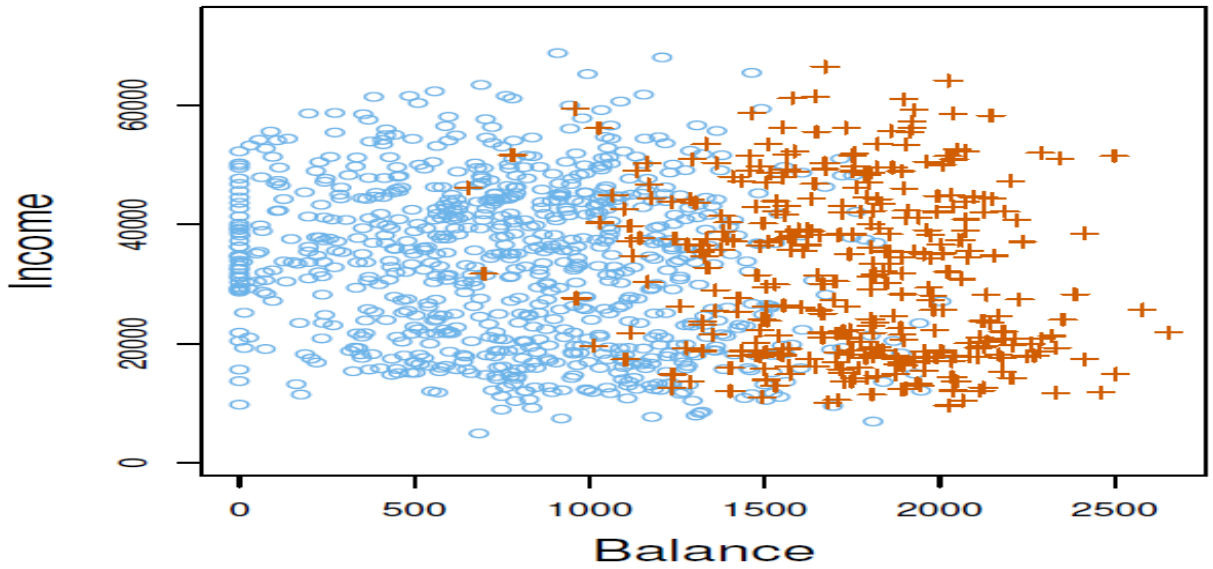


Figure: Données par défaut de la carte de crédit

Exemple 5 : Diagnostics

Dans le domaine de la santé, la phase de diagnostic permet de suivre et d'orienter les patients. De nouvelles techniques permettent au medecin de

- optimiser un bon diagnostic,
- gagner du temps,
- détecter les anomalies sur les images des radios.

Quelques projets exécutés ou en cours

- Amazon a lancé fin 2018 Amazon Comprehend Medical [1], nouveau service dédié aux professionnels de santé. Ce service utilise les techinques de machine learning pour analyser les dossiers médicaux des patients et leur faire gagner du temps dans la prise de décision.
- Le deploiement des assistants virtuels (infirmières virtuelles) dans les hôpitaux de dernière génération. Ces assistants sont capables d'interroger les patients et même repondre aux questions.

[1] https://aws.amazon.com/fr/comprehend/medical/

Exemple 6 : Planning familial

Les données des efforts des plannings familiaux en Amérique du Sud [1].
Le niveau social et les efforts des plannings familiaux sont mesurés par une combinaison d'indices. Plus l'indice est élevé plus le niveau social (resp. l'effort) est élevé.

	Niv. social	Effort	Déclin du tx nat.
Bolivia	46	0	1
Brazil	74	0	10
Chile	89	16	29
Colombia	77	16	25
CostaRica	84	21	29
Cuba	89	15	40
Dominican Rep	68	14	21
Ecuador	70	6	0
El Salvador	60	13	13
⋮	⋮	⋮	⋮

[1] Mauldin and Berelson, 1978

Dans ce problème, on cherche à exprimer le taux de natalité en fonction du niveau social et les efforts de planification. Le but de cette étude est de comprendre comment le niveau social et les efforts de planification influent sur le taux de natalité.

Il s'agit d'un problème de **régression linéaire**.

Dans cette base de données, il existe 20 observations (individus).

- variables explicatives : le niveau social et les efforts de planification.
- variable expliquée : le taux de natalité.

Dans tous les exemples, nous avons

- l'utilisation des données pour construire un **modèle de prédiction** qui sera capable de prédire de nouvelles observations.
- des problèmes d'**apprentissage supervisé**.
  - Apprentissage supervisé : la construction de modèles pour prédire ou estimer un résultat basé sur un ou plusieurs entrées ou fonctionnalités (données labélisées).
  - Apprentissage non supervisé : décrire comment les données sont organisées ou regroupés. C'est-à-dire déterminer les patterns dans les données non labélisées.

Dans un problème d'apprentissage supervisé, nous commençons par une suite composée des observations et de réponses $(\mathbf{X}_i, y_i)_{1 \leq i \leq N}$ $(N \in \mathbb{N})$.

- $\mathbf{X}_i$ sont les vecteurs de variables explicatives (les **prédicteurs** ou **features**). Ces variables peuvent être **qualificatives** (nominales ou ordinales) ou **quantitatives** (discrètes ou continues).
  Notons par $A_{\mathbf{X}}$ l'ensemble de toutes les variables $\mathbf{X}_i$.

- y_i représentent les observations de la variable **expliquée** ou **réponse**. Ces variables peuvent être **catégorielles** avec deux ou plusieurs modalitées ou **quantitatives** (discrètes ou continues).
  Notons par $A_{\mathbf{y}}$ l'ensemble de toutes les variables $\mathbf{y}_i$.

L'objectif principal de l'**apprentissage automatique** (**machine learning** abrégé en **ML**)

- Construire un modèle qui donne les valeurs de la variable **réponse** $y_i \in A_{\mathbf{y}}$ en fonction des **prédicteurs** $\mathbf{X}_i \in A_{\mathbf{X}}$
- La prédiction des observations futures doit être précise.

Définition

Un modèle de **machine learning** est une fonction mathématique définie par

$$\begin{array}{rccc} \hat{f}_N : & A_{\mathbf{X}} & \longrightarrow & A_{\mathbf{y}} \\ & \mathbf{X} & \longrightarrow & \hat{f}_N(\mathbf{X}) \end{array}$$

et permet de prédire le résultat de nouvelles observations.

Meilleur modèle

Un bon modèle (**meilleur modèle**) est celui qui prédit le résultat d'une observation avec précision.

Si $\mathbf{X}_{N+p}$ est une nouvelle observation, le but d'un bon modèle est de prédire la sortie y_{N+p} avec une précision élevée.

Algorithme d'apprentissage

Un **algorithme d'apprentissage** est une fonction définie par

$$\begin{array}{rccc} \mathcal{A}: & \cup_{i\in\mathbb{N}}(A_{\mathbf{X}} \times A_{\mathbf{y}}) & \longrightarrow & F(A_{\mathbf{X}}, A_{\mathbf{y}}) \\ & R & \longrightarrow & \hat{f}_N \end{array}$$

- Un algorithme d'apprentissage construit un modèle de prédiction qui peut être utilisé par la suite pour prédire de nouvelles observations.

- Un algorithme d'apprentissage utilise un ensemble (base de données) d'entraînement (appelée **train set**) pour "apprendre" la relation entre les variables explicatives et la réponse.
- Un ensemble de données (appelé **test set**) est utilisé pour calculer la performance et la précision d'un algorithme d'apprentissage.

Problème de classification :

- les variables explicatives ou **features** sont qualificatives ou quantitatives (discrètes ou continues)
- La réponse est une variable catégorielle dont chaque modalité correspond à une classe.

Régression linéaire :

- La variable réponse y est une variable quantitative.

Dans la suite du cours, nous supposons que $\mathbf{X} \in \mathbb{R}^p$ (p variables explicatives) un vecteur aléatoire de réalisations $(\mathbf{X}_i)_{0 \leq i \leq N}$ et $y \in \mathbb{R}$ une variable aléatoire de réalisations $(y_i)_{0 \leq i \leq N}$.
Les N réalisations $(\mathbf{X}_i, y_i)$ de $(\mathbf{X}, y)$ sont considérées indépendantes et de même loi de distribution $\mathbb{P}$ (En pratique, les données ne sont pas indépendantes et identiquement distribuées $(i.i.d)$).

En général, la fonction mathématique du modèle est donnée par $y = f(\mathbf{X})$ et son estimation nécessite l'écritutre une **fonction de perte** pour minimiser les erreurs de prédiction.

Quelques de fonctions utilisées pour mesurer les erreurs de prédiction :

- La **fonction de perte** l mesure la différence entre la **vraie** valeur de y et la valeur **estimée** $\hat{y}$.

$$\begin{array}{rccc} l: & \mathbb{R} \times \mathbb{R} & \longrightarrow & \mathbb{R}_+ \\ & (y, \hat{y}) & \longrightarrow & l(y, \hat{y}) \end{array}$$

- La **fonction risque** mesure la qualité du modèle f et correspond à la moyenne des pertes.

$$\mathcal{R}(f) = \mathbb{E}(l(y, f(\mathbf{X})) = \int l(y, f(\mathbf{x}))d\mathbb{P}(x, y)$$

- La **perte quadratique** :

$$l_q(y, f(\mathbf{X})) = \Big(y - f(\mathbf{X})\Big)^2$$

Sommaire

Problème de score :

Soit **X** les caractéristiques des emprunteurs (variables explicatives ou exogènes). Notons par $s(\mathbf{X})$ le score qui sert à évaluer la probabilité que l'emprunteur soit en **défaut** (noté D). Nous notons

- deux classes de **prédiction** : la classe **y=0** correspond aux bons emprunteurs et **y=1** correspond aux mauvais emprunteurs
- deux classes d'**observation** : la classe D des emprunteurs en défaut et la classe ND des emprunteurs en survie

Pour un modèle de prédiction parfait, nous retrouvons

- tous les éléments de la classe **y=1** observés dans la classe D
- tous les éléments de la classe **y=0** observés dans la classe ND

Les varaibles explicatives ou endogènes de ce problème :

- ratios financiers (revenus, charges, niveau d'endettement, etc)
- caractéristiques socio-économiques (âge, statut marital,etc)
- performance des crédits passés
- nature des prêts souscrits

Définition

Soit $(\Omega, \mathcal{F}, \mathbb{P})$ un espace probabilisé. Soient A et B deux évènements tels que $\mathbb{P}(B) \neq 0$. On définit la probabilité de A sachant B par

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

Exemple : Portefeuille de 1150 prêts immobiliers de la banque de Vinci.

Statut	Nbre_ND	Nbre_D
Propriétaire	600	30
Locataire	200	70
Investisseur	225	25
Total	1025	125

$$\mathbb{P}(D) = \frac{125}{1025 + 125} = 0.109, \quad \mathbb{P}(ND) = \frac{1025}{1025 + 125} = 0.891$$

$$\mathbb{P}(ND|Locataire) = \frac{200}{270} = 0.74$$

La probabilité pour que la réponse y d'appartenne à une classe $m \in \{0, 1\}$ sachant la valeur de la variable explicative $\mathbf{X}$, peut être déterminée avec le théorème de Bayes (probabilité conditionnelle).

Théorème de Bayes

$$\mathbb{P}(y = m|\mathbf{X} = x) = \frac{\mathbb{P}(y = m)}{\mathbb{P}(\mathbf{X} = x)}.\mathbb{P}(\mathbf{X} = x|y = m)$$

Dans le cas de la regression logistique, la probabilité $\mathbb{P}(y = m|\mathbf{X} = x)$ correspond à une fonction logistique. La regression logistique n'est pas souvent recommendée pour les raisons suivantes

- L'**estimateur du maximum de vraisemblance** (fonction coût) de la fonction logistique ne converge pas (données separables). On peut envisager l'analyse discriminante.
- Si le nombre d'échantillon est petit et la distribution de $\mathbf{X}$ est approximativement normale dans chaque classe de y , l'analyse discriminante est plus stable que la régression logistique.

L'analyse discriminante peut aussi être envisagée lorsqu'il y'a plus de deux classes.

Supposons que nous disposons $M \geq 2$ classes.

- La probabilité **à priori** de la réponse y de la classe $m \in \{1, 2, \cdots, M\}$ est notée $\pi_m = \mathbb{P}(y = m)$
- La densité conditionnelle $f_m(x)$ de la variable $\mathbf{X}$ sachant que $y = m$.

Si $x \in \mathbb{R}^p$ alors $f_{\mathbf{X}}(x) = \prod_{m=1}^{M} \pi_m f_m(x)$

Illustration : deux lois normales dont les densités sont

$f_1 \sim \mathcal{N}\left(\begin{pmatrix}1\\2\end{pmatrix}, \begin{pmatrix}2 & 0\\0 & 2\end{pmatrix}\right)$ et $f_2 \sim \mathcal{N}\left(\begin{pmatrix}7\\5\end{pmatrix}, \begin{pmatrix}1 & 0\\0 & 3\end{pmatrix}\right)$.

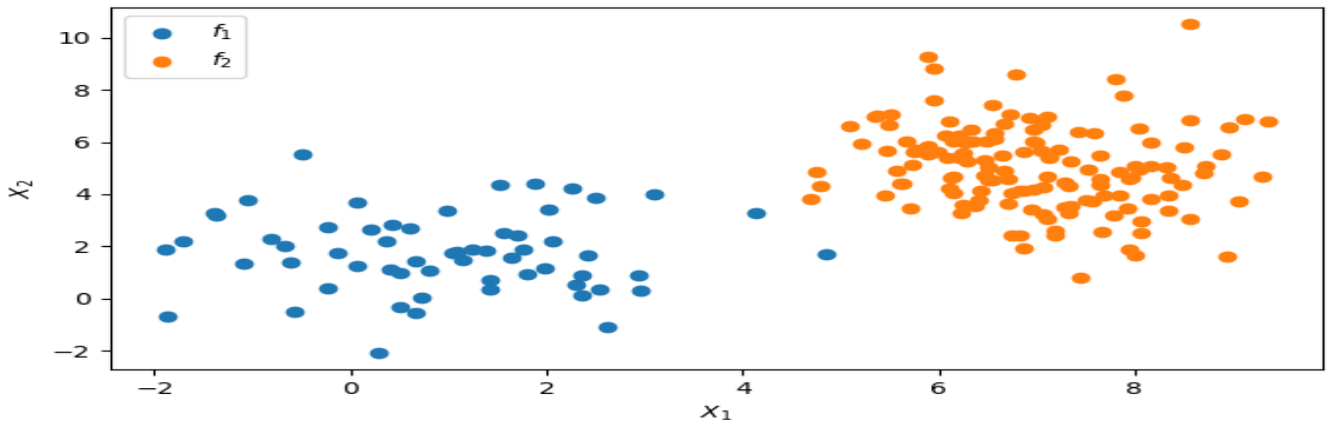


Figure: Simulation de deux classes

Nous avons

$$\pi_1 = 0.3, \pi_2 = 0.7, f_{\mathbf{X}}(x) = 0.3 f_1(x) + 0.7 f_2(x)$$

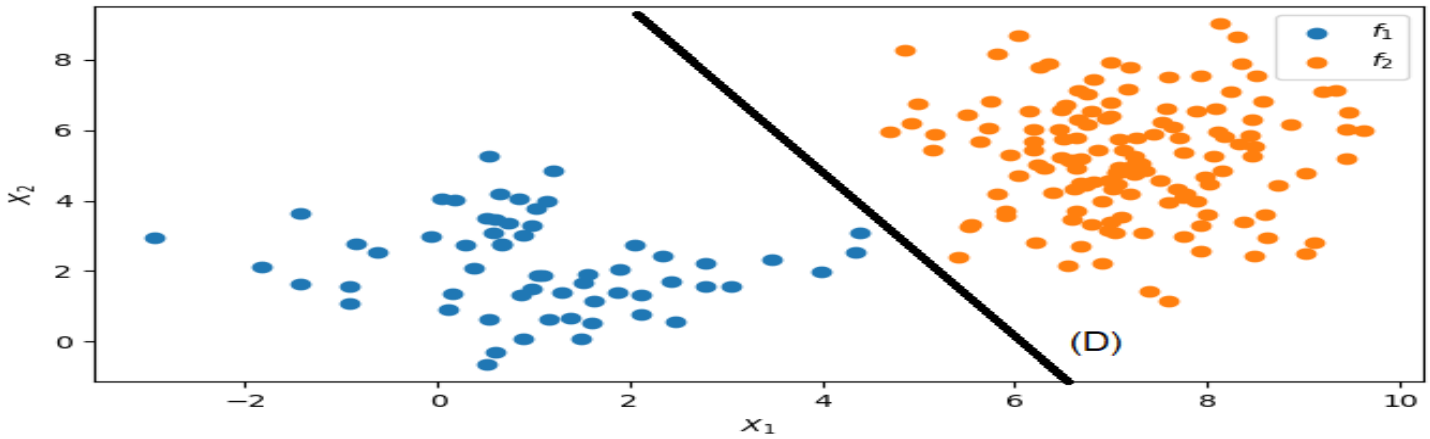


Figure: Classifier (D)

La règle de classification est donnée par

$$\mathcal{C}(x) = \begin{cases} 1 & si \quad x \text{ est à gauche de } (D) \\ 2 & si \quad x \text{ est à droite de } (D) \end{cases}$$

Définition

Un classifier $\mathcal{C}$ est une fonction mesurable définie sur $A_{\mathbf{X}}$ à valeurs dans A_y

Soit $\mathbf{X}_{N+1}$ une nouvelle observation. Le **classifier** (ou encore la **règle de décision**) désigne la classe à laquelle appartient cette observation en calculant $\mathcal{C}(\mathbf{X}_{N+1})$

A partir de la définition et le rôle d'un classifier, plusieurs questions peuvent se poser.

- Comment construire un classifier à partir d'un ensemble de données d'apprentissage (**train set**)?
- Comment évaluer la qualité d'un classifier?
- Pouvons nous déterminer un classifier **optimal**?

Pour répondre à certaines questions, nous avons besoin d'une **fonction perte** pour le calcul de l'erreur des observations mal classées.
La **fonction perte** $l(m_1, m_2)$ représente l'erreur lorsque $y = m_1$ alors que le classifier donne $y = m_2$. Elle est définie par

$$l(m_1, m_2) = 1_{m_1 \neq m_2} = \begin{cases} 0 & si \quad m_1 = m_2 \\ 1 & si \quad m_1 \neq m_2 \end{cases}$$

Soit $\mathcal{C}$ un classifier. Le risque de $\mathcal{C}$ est donné par

$$\mathcal{R}(\mathcal{C}) = \mathbb{E}(l(y, \mathcal{C}(\mathbf{X}))) = \int_{A_{\mathbf{X}} \times A_y} l(y, \mathcal{C}(\mathbf{X})) d\mathbb{P}(x, y) = \mathbb{P}(y \neq c(\mathbf{X})).$$

Puisque l appartient à $\{0, 1\}$, alors un **meilleur** classifier est celui avec le risque $\mathcal{R}(\mathcal{C})$ minimal.

la fonction perte (formule (1)) n'est pas toujours appropriée à tous les problèmes de classification (problème mail/spam).

Définition

La **probabilité a posteriori** de la classe $y = m$ sachant que $\mathbf{X} = x$ est

$$\mathbb{P}(y = m | \mathbf{X} = x) = \frac{\pi_m f_m(x)}{f_{\mathbf{X}}(x)} = \frac{\pi_m f_m(x)}{\prod_{k=1}^{M} \pi_k f_k(x)}.$$

Il s'agit de la probabilité qu'une observation avec une valeur prédictive x appartienne à la classe $y = m$.

Naturellement, l'observation x appartient à la classe $y = m$ si la valeur de la probabilité $\mathbb{P}(y = m|\mathbf{X} = x)$ est large.

Classifier Bayesien

Un classifier Bayesien $\mathcal{C}^*$ attribue à une observation la classe ayant la plus grande probabilité sachant la valeur x de l'observation.

$$\mathcal{C}^*(x) = m \;\; si \;\; \mathbb{P}(y = m|\mathbf{X} = x) = \max_{k \in \{1,2,\cdots,M\}} \mathbb{P}(y = k|\mathbf{X} = x)$$

$$\Leftrightarrow \mathcal{C}^*(x) = arg \max_{k \in \{1,2,\cdots,M\}} \mathbb{P}(y = k|\mathbf{X} = x)$$

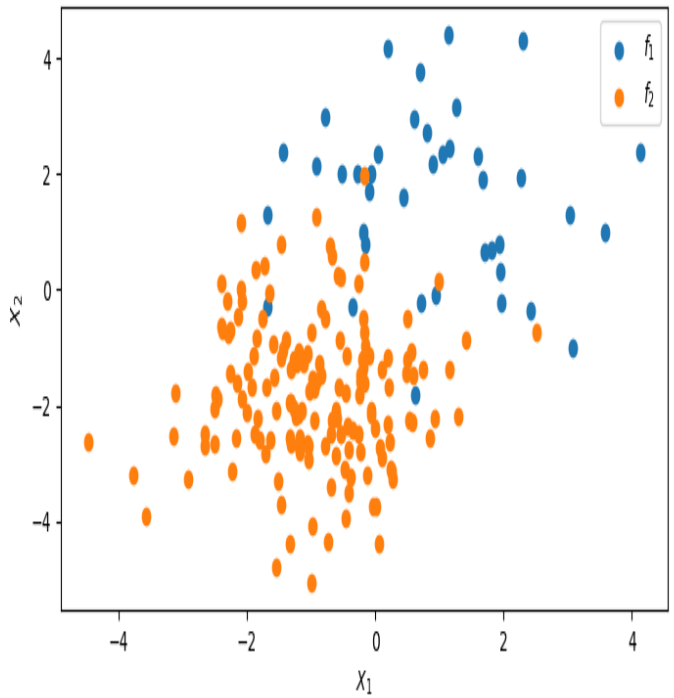
Remarque :

- Pour un problème à deux classes (0–1). Le classifier Bayesien prédit la classe $y = 0$ si $\mathbb{P}(y = 0|\mathbf{X} = x) > 0.5$.
- Si la densité $f_{\mathbf{X}}(x)$ est indépendante des classes alors le classifier peut se réécrire comme $\mathcal{C}^*(x) = arg \max_{k \in \{1,2,\cdots,M\}} \pi_k f_k(x)$

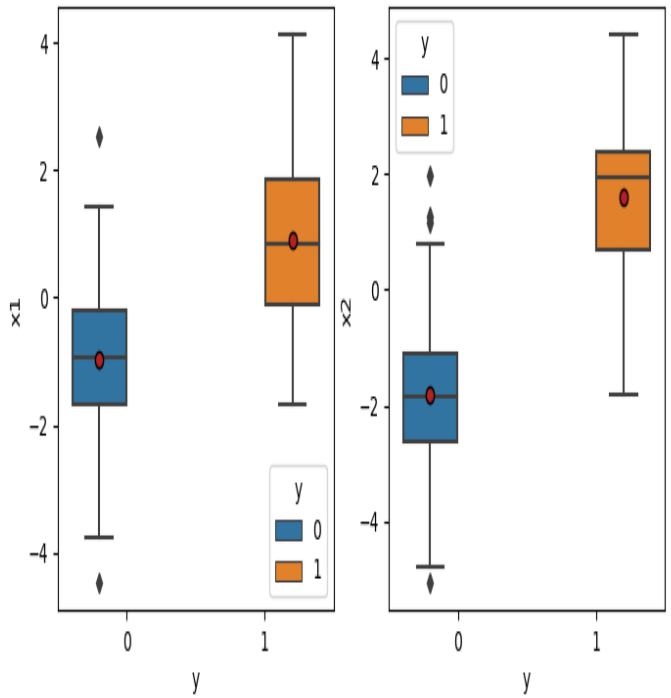
Illustration

deux lois normales dont les densités ($\pi_1 = 0.2, \pi_2 = 0.8$) sont

$f_1 \sim \mathcal{N}\left(\begin{pmatrix}1\\2\end{pmatrix}, \begin{pmatrix}2 & 0\\0 & 2\end{pmatrix}\right)$ et $f_2 \sim \mathcal{N}\left(\begin{pmatrix}-1\\-2\end{pmatrix}, \begin{pmatrix}1 & 0\\0 & 2\end{pmatrix}\right)$



(a) : Simulation de deux classes



(b) : Boxplots des deux variables

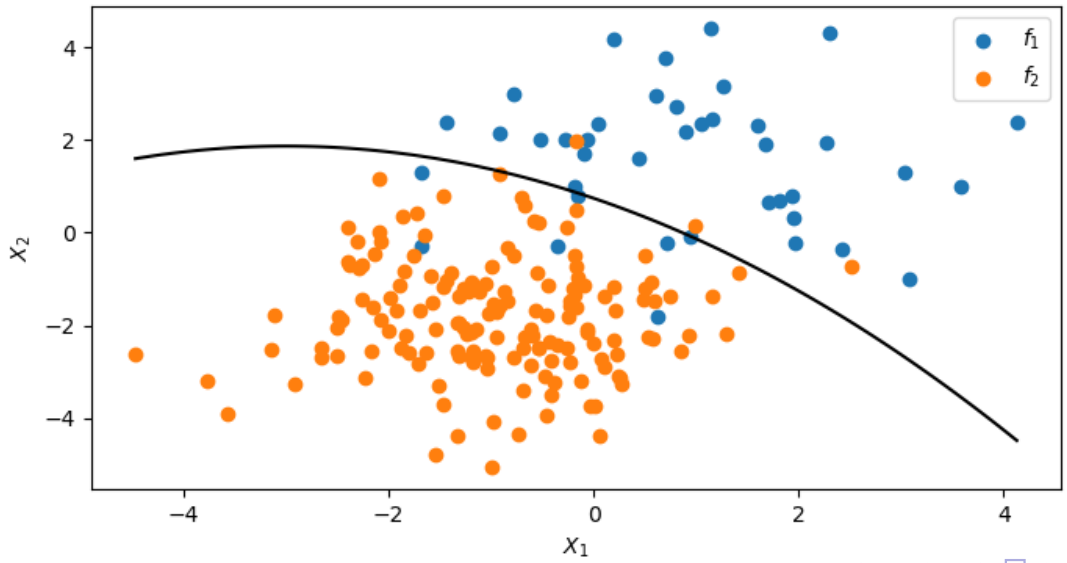
Les densités des deux classes :

$$f_{y=1}(\mathbf{X} = (x_1, x_2)) = \frac{1}{4\pi} \exp\Big(-\frac{1}{4}((x_1 - 1)^2 + (x_2 - 2)^2)\Big),$$

$$f_{y=2}(\mathbf{X} = (x_1, x_2)) = \frac{1}{2\sqrt{2}\pi} \exp\Big(-\frac{1}{2}((x_1 + 1)^2 + \frac{1}{2}(x_2 + 2)^2)\Big).$$

La densité de $\mathbf{X}$: $f_{\mathbf{X}}(x_1, x_2) = 0.2 f_{y=1}(x_1, x_2) + 0.8 f_{y=2}(x_1, x_2)$ La frontière des deux classes est déterminée en posant

$$0.2 f_{y=1}(x_1, x_2) = 0.8 f_{y=2}(x_1, x_2) \Leftrightarrow x_1^2 + 6x_1 + 8x_2 = 4\ln(4\sqrt{2}) - 1.$$



- La courbe noire représente les points $\mathbf{X} = (x_1, x_2)$ tels que la probabilité d'appartenir est égale à 0.5. C'est la **frontière de la décision de Bayes**.
- Les points à droite de la courbe noire ont une probabilité supérieure à 0.5 tandis que ceux à gauche ont une probabilité inférieure à 0.5.

Proposition

Parmis tous les classifiers, le classifier Bayesien est le moins risqué. Il est dit optimal.

Preuve : Soit $\mathcal{C}$ un classifier, nous avons

$$\mathcal{R}(\mathcal{C}) = \mathbb{E}(I(y, \mathcal{C}(\mathbf{X}))) = \mathbb{E}\Big(\mathbb{E}(I(y, \mathcal{C}(\mathbf{X})|\mathbf{X})\Big) = \int \mathbb{E}(I(y, \mathcal{C}(\mathbf{X})|\mathbf{X}) f_{\mathbf{X}}(x) dx$$

Soit $\mathcal{C}^*$ qui minimise $\mathbb{E}(I(y, \mathcal{C}(\mathbf{X})|\mathbf{X})$ alors

$$\mathbb{E}(I(y, \mathcal{C}^*(\mathbf{X})|\mathbf{X}) \leq \mathbb{E}(I(y, \mathcal{C}(\mathbf{X})|\mathbf{X})$$

Puisque $\forall x \in \mathbb{R}^p$, $f_{\mathbf{X}}(x) \geq 0$ alors $\mathcal{R}(\mathcal{C}^*) \leq \mathcal{R}(\mathcal{C})$

Nous avons

$$\mathbb{E}(l(y, \mathcal{C}(\mathbf{X})|\mathbf{X}) = \sum_{m=1}^{M} l(y = m, \mathcal{C}(x))\mathbb{P}(y = m|\mathbf{X} = x)$$

Si $\mathcal{C}(x) = m'$ alors

$$\begin{aligned}\mathbb{E}(l(y, \mathcal{C}(\mathbf{X})|\mathbf{X})) &= \sum_{m=1}^{M} l(y = m, m')\mathbb{P}(y = m|\mathbf{X} = x) \\ &= \sum_{m=1}^{M} 1_{m \neq m'}\mathbb{P}(y = m|\mathbf{X} = x) \\ &= \sum_{m \neq m'} \mathbb{P}(y = m|\mathbf{X} = x) = 1 - \mathbb{P}(y = m'|\mathbf{X} = x)\end{aligned}$$

$$\underset{k \in \{1,2,\cdots,M\}}{arg\,min} \Big(1 - \mathbb{P}(y = k|\mathbf{X} = x)\Big) = \underset{k \in \{1,2,\cdots,M\}}{arg\,max} \mathbb{P}(y = k|\mathbf{X} = x)$$

qui correspond au classifier Bayesien alors $\mathcal{C}^*$ est le classifier Bayesien.

Remarque : l'optmalité du classifier Bayesien n'implique pas que le risque est petit.

Illustration

Premier cas : $f_1 \sim \mathcal{N}(-1, 0.5)$, $f_2 \sim \mathcal{N}(1, 0.5)$ et $\pi_1 = \pi_2$. Le classifier Bayesien est donné par

$$\mathcal{C}^*(x) = \begin{cases} 1 & si \quad f_1(x) > f_2(x) \\ 2 & si \quad f_1(x) < f_2(x) \end{cases}$$

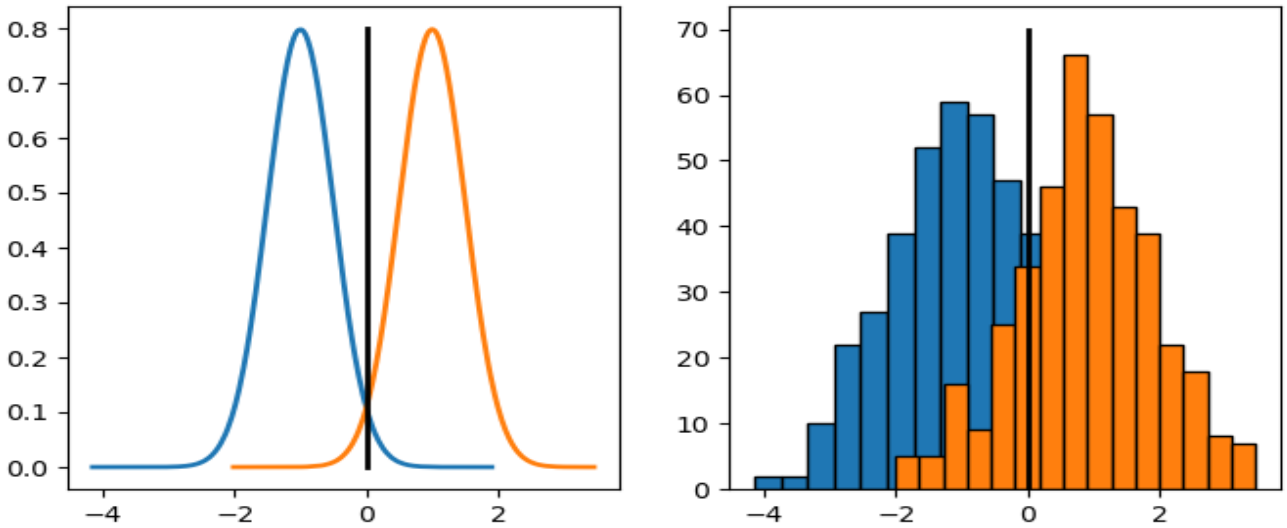


Figure: (a): Densités des deux lois. (b) Histogrammes des deux lois. La courbe noire représente le classifier.

Second cas : $f_1 \sim \mathcal{N}(-0.5, 1)$, $f_2 \sim \mathcal{N}(0.5, 1)$ et $\pi_1 = \pi_2$

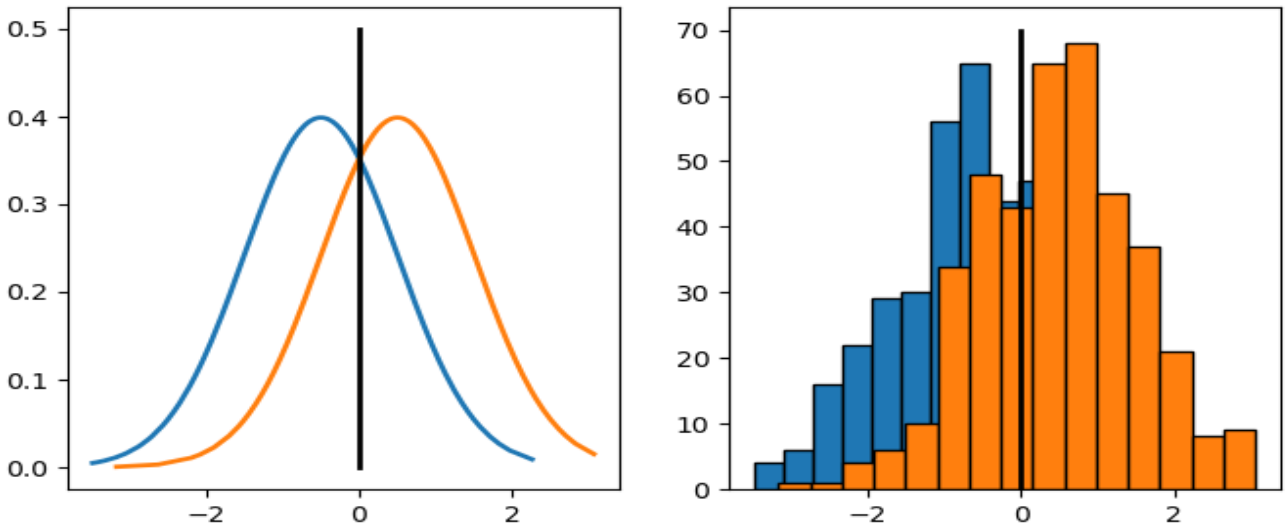


Figure: (a): Densités des deux lois. (b) Histogrammes des deux lois. La courbe noire représente le classifier.

Supposons que la variable explicative $\mathbf{X}$ suit une loi normale multidimensionnelle (à plusieurs variables) de centre (vecteur moyenne) $\mu \in \mathbb{R}^p$ et de matrice de variance-covariance $\Sigma \in \mathcal{M}_p(\mathbb{R})$. La matrice Σ est semi-définie positive. La fonction de densité de $\mathbf{X}$:

$$f_{\mathbf{X}}(x) = \frac{1}{(2\pi)^{\frac{p}{2}}\sqrt{det(\Sigma)}} \exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big), \quad x \in \mathbb{R}^p$$

Supposons que la variable explicative $\mathbf{X}$ suit une loi normale multidimensionnelle (à plusieurs variables) de centre (vecteur moyenne) $\mu \in \mathbb{R}^p$ et de matrice de variance-covariance $\Sigma \in \mathcal{M}_p(\mathbb{R})$. La matrice Σ est semi-définie positive. La fonction de densité de $\mathbf{X}$:

$$f_{\mathbf{X}}(x) = \frac{1}{(2\pi)^{\frac{p}{2}}\sqrt{det(\Sigma)}} \exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big), \quad x \in \mathbb{R}^p$$

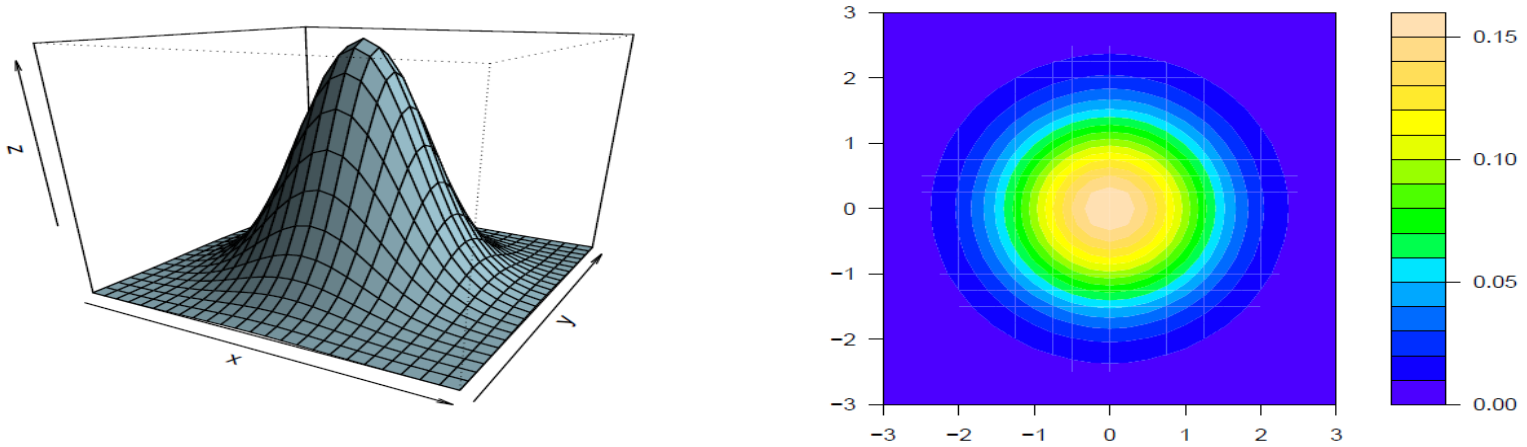


Figure: Fonction de densité $f_{\mathbf{X}}$ pour $\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ et $\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

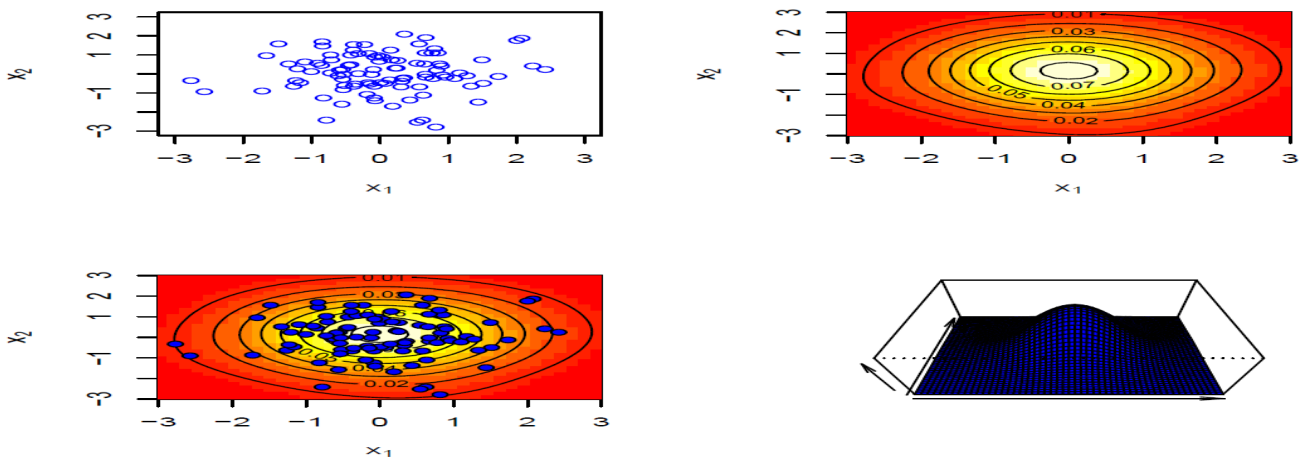


Figure: Représentations graphiques des données générées avec la loi $\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}\right)$

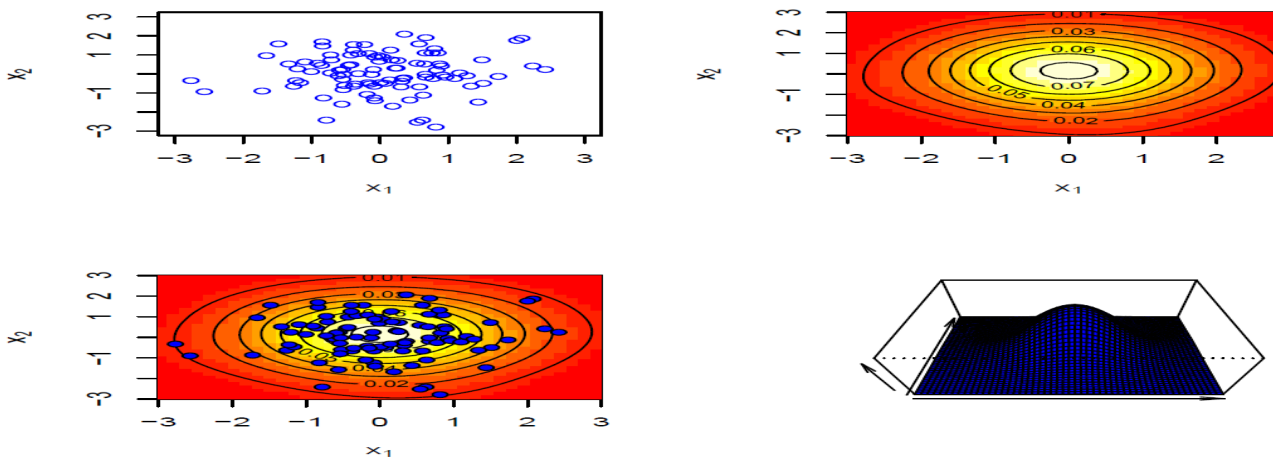


Figure: Représentations graphiques des données générées avec la loi $\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}1 & 0\\0 & 1\end{pmatrix}\right)$

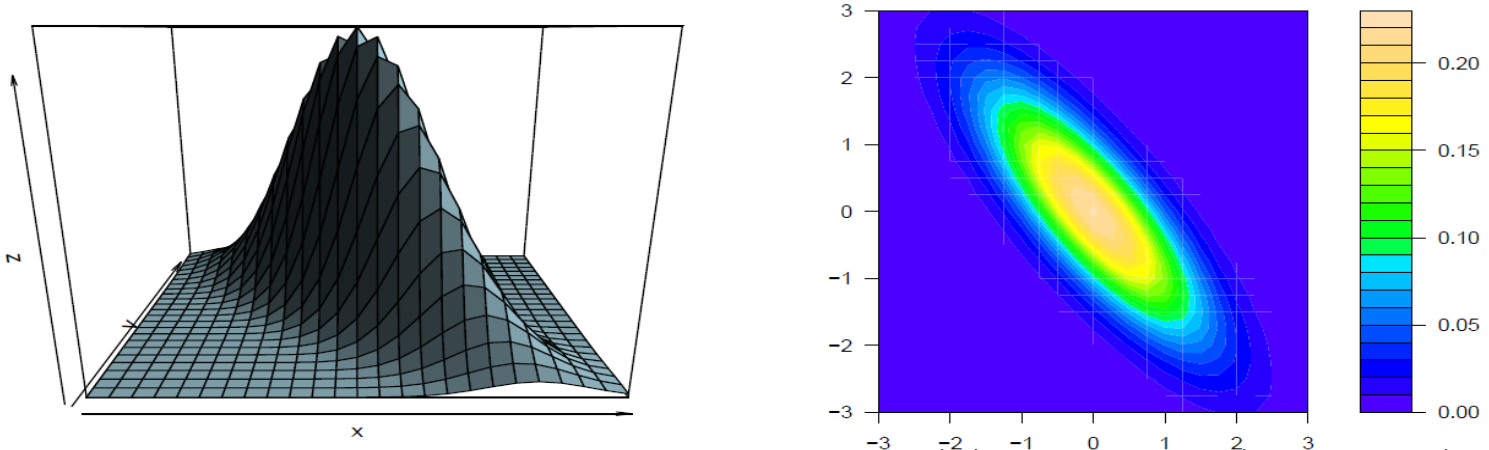


Figure: Fonction de densité $f_{\mathbf{X}}$ pour $\mu = \begin{pmatrix}0\\0\end{pmatrix}$ et $\Sigma = \begin{pmatrix}1 & -1\\-1 & 1.5\end{pmatrix}$

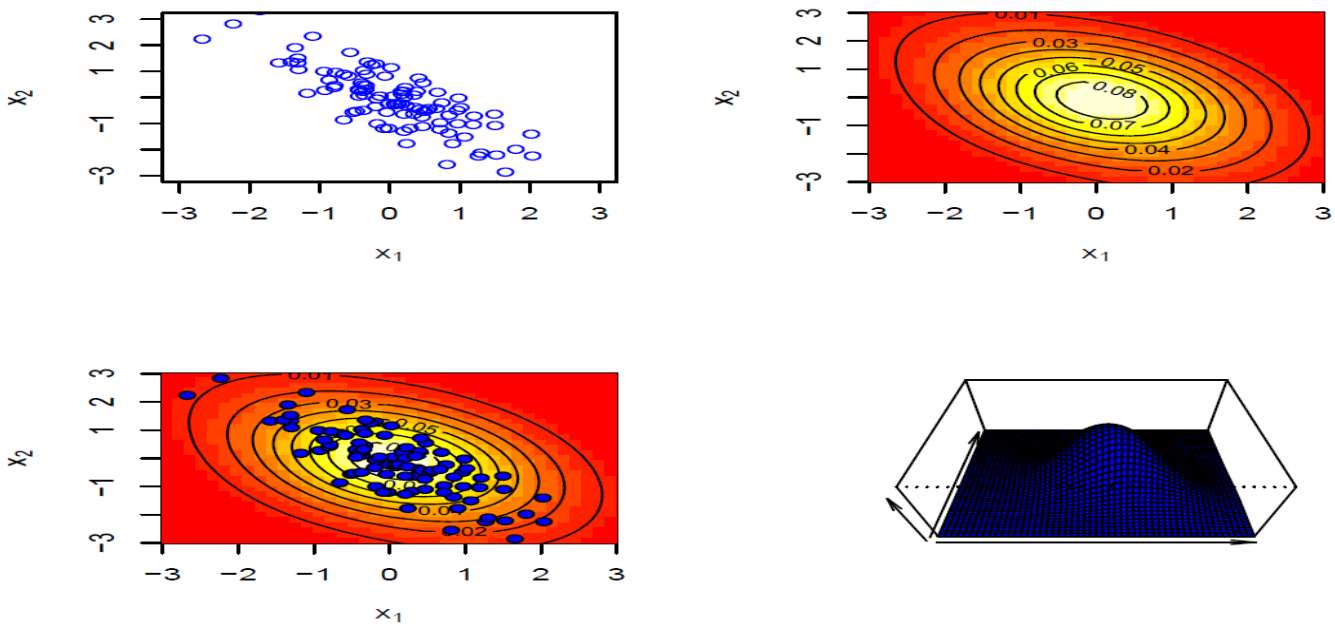


Figure: Représentations graphiques des données générées avec $\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}1 & -1\\-1 & 1.5\end{pmatrix}\right)$

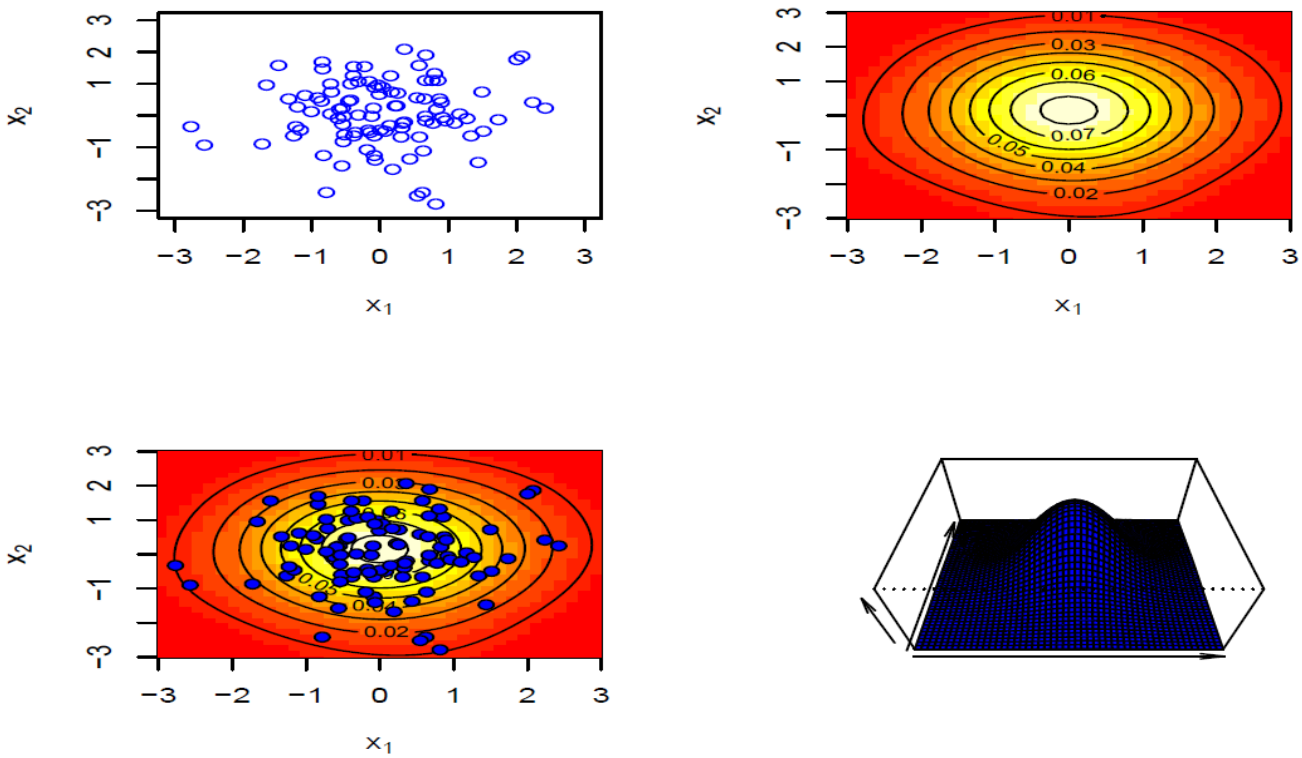


Figure: Simulation de deux classes de lois gaussiennes multidimensionnelles

Les données de la classe **y=m** suit une loi gaussienne multidimensionnelle de paramètres μ_m et Σ (identique pour toutes les classes). La densité de la classe :

$$f_m(x) = \frac{1}{(2\pi)^{\frac{p}{2}}\sqrt{det(\Sigma)}} \exp\Big(-\frac{1}{2}(x-\mu_m)^T\Sigma^{-1}(x-\mu_m)\Big), \quad x \in \mathbb{R}^p$$

La probabilité de prédiction de la classe **y=m** sachant la valeur de **X** :

$$\mathbb{P}(y = m|\mathbf{X} = x) = \frac{\pi_m f_m(x)}{f_{\mathbf{X}}(x)} = \frac{\pi_m f_m(x)}{\prod_{k=1}^{M} \pi_k f_k(x)}$$

Le classifier Bayesien attribue l'observation $\mathbf{X} = x$ à la classe dont

$$\begin{aligned} C^*(x) &= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ \pi_k f_k(x) = \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ ln(\pi_k) + ln(f_k(x)) \\ &= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ ln(\pi_k) - \frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) \\ &= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ ln(\pi_k) + \mu_k^T\Sigma^{-1}x - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k^T \\ &:= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ \delta_k^L(x). \end{aligned}$$

Les frontières de la décision de Bayes sont déterminées en posant $\delta_{m_1}^L(x) = \delta_{m_2}^L(x)$ pour tout $m_1 \neq m_2$. Ces frontières séparent les données en M domaines.

Exemple :

$\mu_1 = \begin{pmatrix} -2 \\ 6 \end{pmatrix}$, $\pi_1 = 0.3$, $\mu_2 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$, $\pi_2 = 0.5$, $\mu_3 = \begin{pmatrix} 6 \\ 6 \end{pmatrix}$, $\pi_2 = 0.2$ et $\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Les fonctions qui séparent les trois classes sont donnnées par :

$$\delta_1^L(x_1, x_2) = ln(\pi_1) - 2x_1 + 6x_2 - 20, \delta_2^L(x_1, x_2) = ln(\pi_2) + 2x_1 + 2x_2 - 4,$$
$$\delta_3^L(x_1, x_2) = ln(\pi_3) + 6x_1 + 6x_2 - 36$$

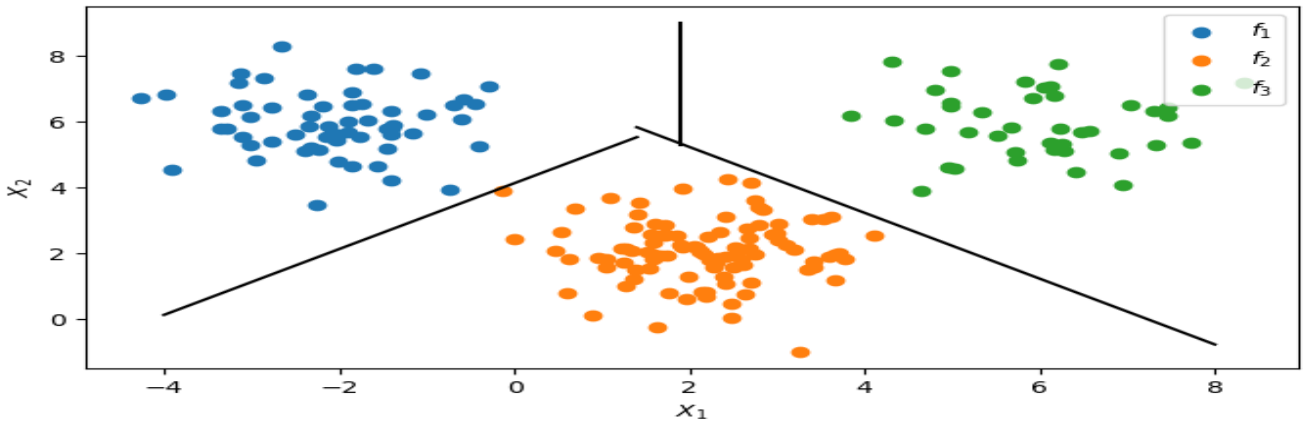


Figure: Exemple de trois classes de données gaussiennes et le classifier de décision de Bayes en noir.

En pratique, il faudra vérifier la normalité de la variable explicative **X** et les estimations des paramètres se vont avec l'échantillon d'apprentissage. Cela correspond à

$$\hat{\mu}_m = \frac{1}{N_m} \sum_{j=1}^{N_m} x_j, \tag{2}$$

$$\hat{\Sigma} = \frac{1}{N - M} \sum_{k=1}^{M} \sum_{i:y_i=k} (x_i - \mu_k)(x_i - \mu_k)^T,$$

$$\hat{\pi}_m = \frac{N_m}{N}; \tag{3}$$

Avec N_m le nombre d'éléments dans la classe $y = m$ ($\sum_{k=1}^{M} N_k = N$), M le nombre de classes.

L'analyse discriminante linéaire (**LDA**) attribue à l'observation $\mathbf{X} = x$ la classe définie ci-dessous.

$$\begin{aligned} \mathbf{LDA}(x) &= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\; ln(\pi_k) + \mu_k^T \hat{\Sigma}^{-1} x - \frac{1}{2}\hat{\mu}_k^T \hat{\Sigma}^{-1} \hat{\mu}_k^T \\ &:= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\; \hat{\delta}_k^L(x). \end{aligned}$$

- La fonction **LDA** est une fonction affine en x et linéaire par rapport à ces paramètres.
- L'utilisation de la fonction **LDA** suppose que les données dans chaque classe suivent une loi gaussienne de centre μ_k lié à la classe. Toutes les classes ont la même matrice variance-covariance.

NB : Le classiifier **LDA** n'est pas pertinente lorsque les matrices variance-covariance des classes sont différentes.

Supposons que la variable explicative de chaque classe $y = m$ suit une loi normale multidimensionnelle de centre $\mu_m \in \mathbb{R}^p$ et de matrice de variance-covariance $\Sigma_m \in \mathcal{M}_p(\mathbb{R})$. La fonction de densité de $\mathbf{X}$ est donnée par

$$f_m(x) = \frac{1}{(2\pi)^{\frac{p}{2}}\sqrt{det(\Sigma_m)}} \exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma_m^{-1}(x-\mu)\Big), \quad x \in \mathbb{R}^p$$

La probabilité de prédiction de la classe $y = m$:

$$\mathbb{P}(y = m|\mathbf{X} = x) = \frac{\pi_m f_m(x)}{f_{\mathbf{X}}(x)} = \frac{\pi_m f_m(x)}{\prod_{k=1}^{M}\pi_k f_k(x)}$$

Le classifier Bayesien attribue $\mathbf{X} = x$ à la classe dont

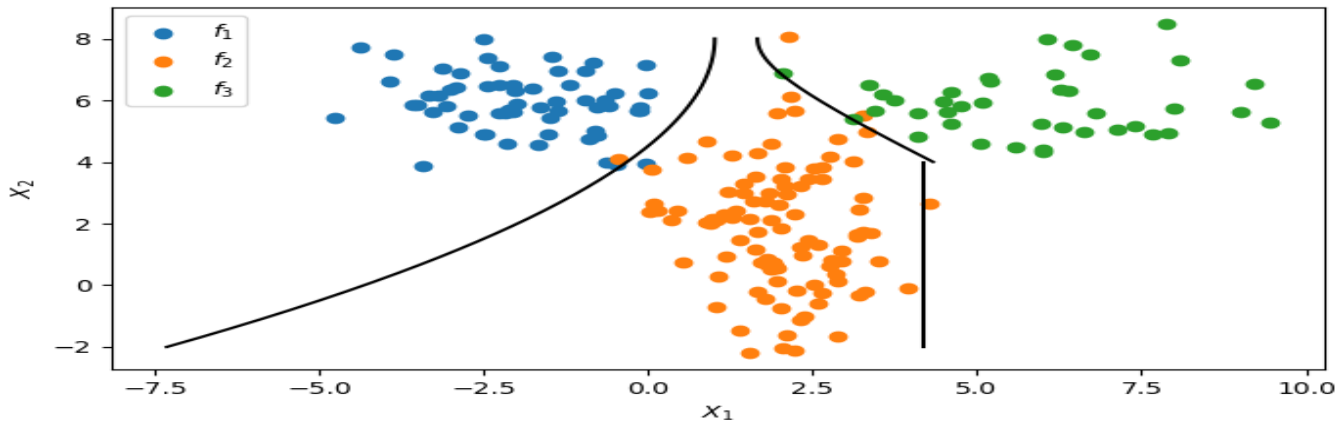
$$\mathcal{C}^*(x) = \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ \pi_k f_k(x) = \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\ ln(\pi_k) + ln(f_k(x))$$

$$\begin{aligned} C^*(x) &= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\; ln(\pi_k) - \frac{1}{2}ln(det(\Sigma_k)) - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu k) \\ &:= \underset{k\in\{1,2,\cdots,M\}}{arg\,max}\; \delta_k^Q(x) \end{aligned}$$

Les frontières de la décision de Bayes sont déterminées en posant $\delta_{m_1}^Q(x) = \delta_{m_2}^Q(x)$ pour tout $m_1 \neq m_2$. Ces frontières séparent les données en M domaines.

Exemple : $\mu_1 = \begin{pmatrix} -2 \\ 6 \end{pmatrix}, \Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \pi_1 = 0.3,$

$\mu_2 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \Sigma_2 = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}, \pi_2 = 0.5, \mu_3 = \begin{pmatrix} 6 \\ 6 \end{pmatrix}, \Sigma_3 = \begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}, \pi_2 = 0.2.$



Lorsque les paramètres des lois des données des classes sont inconnus, on peut estimer μ_m et π_m avec les formules (2) et (3) données dans le cas de l'analyse discriminante linéaire. Les estimations des matrices variance-covariance :

$$\hat{\Sigma}_m = \frac{1}{N-1} \sum_{i:y_i=m} (x_i - \hat{\mu}_m)^T (x_i - \hat{\mu}_m).$$

Le classifier d'analyse discriminante quadratique **QDA** attribue à l'observation $\mathbf{X} = x$ la classe suivante

$$\begin{aligned} \mathbf{QDA}(x) &= \arg\max_{k \in \{1,2,\cdots,M\}} ln(\pi_k) - \frac{1}{2} ln(det(\Sigma_k)) \\ &\quad -\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1} (x - \mu k) \\ &:= \arg\max_{k \in \{1,2,\cdots,M\}} \delta_k^Q(x) \end{aligned}$$

La fonction du classifier $\mathbf{QDA}(x)$ est quadratique en x.

Quelques points importants :

- La préférence de **QDA** à **LDA** ou vice-versa est liée à un compromis entre le bias et la variance.
- Puisque la matrice variance-covariance est symétrique alors dans le cas d'une variable explicative de p composantes, son estimation avec **LDA** nécessite le calcul de $\frac{p(p+1)}{2}$ paramètres.

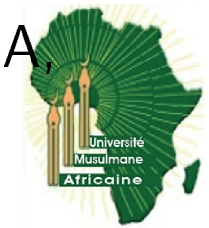
  Le nombre de paramètres pour **QDA** devient $M\frac{p(p+1)}{2}$ où M est le nombre total de classes.
- Le classsifier **LDA** nécessite d'estimer moins de paramètres par rapport à **QDA** et a une variance nettement inférieure. Ce qui peut conduire à une amélioration de performance dans les prédictions.
- Le classifier **LDA** peut avoir des problèmes de biais alors qu'il faut un compromis entre le bias et la variance pour un bon classifier.

Recommandations :

- On peut préférer **LDA** à **QDA** quand il y en a relativement peu observations d'entraînement (et donc la réduction de la variance est cruciale).
- **QDA** est recommandé si l'ensemble de formation est très vaste ou si l'hypothèse d'une matrice de covariance commune est clairement intenable.

Remarques :

- les performances de LDA/QDA peuvent être évaluées à l'aide de la matrice de confusion, de la sensibilité (**sensitivity**) et de la spécificité (**specificity**)
- La courbe **ROC** et l'**AUC** s'appliquent également à la **LDA/QDA** et peuvent être utilisés pour comparer les classificateurs (LDA, QDA, régression logistique).



Sommaire

Exemple d'application

Une chaine de magasin a mis en place une carte de crédit. Elle dispose de 145 clients dont 40 ont connu des défauts de paiement. Les caractéristiques connues des clients sont

- le sexe,
- le taux d'endettement,
- les revenus mensuels,
- les dépenses éffectuées sur les gammes de produit.

Problème

Nous souhaitons savoir si un nouveau client connaîtra des défauts de paiement (prédiction).

Nous disposons de deux classes de prédiction : **y=1** quand le client est en défaut de paiement et **y=0** dans le cas contraire.
La variable y suit une loi **binomiale** de paramètres N et π où N est le nombre d'observations et $\pi = \mathbb{P}(y = 1)$.
La probabilité d'appartenance à la classe **y=0** est

$$\mathbb{P}(y = 0) = 1 - \pi,$$

et nous résumons que pour tout $y_i \in \{0, 1\}$.

$$\mathbb{P}(y = y_i) = \pi^{y_i}(1 - \pi)^{1-y_i}, \text{ avec } i \in \{0, 1, \cdots, N\}$$

Nous rappelons que

$$\mathbb{E}(y) = \pi, \quad Var(y) = \pi(1 - \pi).$$

Question

Comment estimer la probabilité π?

L'estimation de la probabilité π peut se faire avec la méthode du **maximum de vraisemblance**.

Supposons les réalisations de la variable y notées $y_1, y_2, \cdots, y_N$ sont indépendantes et identiquement distribuées.
La **vraisemblance** de π est donnée par

$$L_N(\pi) = \prod_{i=1}^{N} \pi^{y_i}(1-\pi)^{1-y_i}.$$

La **log-vraisemblance** est définie par

$$\mathcal{L}_N(\pi) = \sum_{i=1}^{N} \Big(y_i \log(\pi) + (1-y_i)\log(1-\pi) \Big).$$

Il faut retenir que

$$\max_{\pi} L_N(\pi) = \max_{\pi} \mathcal{L}_N(\pi).$$

La condition du **premier ordre** nous donne

$$\frac{\partial \mathcal{L}_N}{\partial \pi}\Big|_{\pi=\hat{\pi}} = \sum_{i=1}^{N} \left(\frac{y_i}{\pi} - \frac{1-y_i}{1-\pi} \right)\Big|_{\pi=\hat{\pi}} = 0 \Rightarrow \hat{\pi} = \frac{1}{N}\sum_{i=1}^{N} y_i = \bar{y}.$$

La loi faible des grands nombres garantit que $\hat{\pi} \xrightarrow{\mathbb{P}} \mathbb{E}(y) = \pi$ quand N tend vers ∞.
A partir du théorème central limite, nous avons

$$\sqrt{n}\frac{\hat{\pi} - \pi}{\sqrt{\pi(1-\pi)}} = \sqrt{n}\frac{\hat{\pi} - \mathbb{E}(y)}{\sqrt{Var(y)}} \xrightarrow{\mathbb{L}} \mathcal{N}(0,1).$$

A partir du théorème de Slutsky, on a

$$\sqrt{n}\frac{\hat{\pi} - \pi}{\sqrt{\hat{\pi}(1-\hat{\pi})}} \xrightarrow{\mathbb{L}} \mathcal{N}(0,1).$$

L'intervalle de confiance de l'estimateur avec un niveau de risque 5% (Normalité asymptotique de $\hat{\pi}$) :

$$\left[\hat{\pi} - 1.96\sqrt{\frac{\sqrt{\hat{\pi}(1-\hat{\pi})}}{N}}; \hat{\pi} + 1.96\sqrt{\frac{\sqrt{\hat{\pi}(1-\hat{\pi})}}{N}}\right].$$

En faisant la représentation graphique des fréquences des observations des classes en fonctions des variables individuelles, nous remarquons les courbes tendent des fonctions **sigmoïdes**.

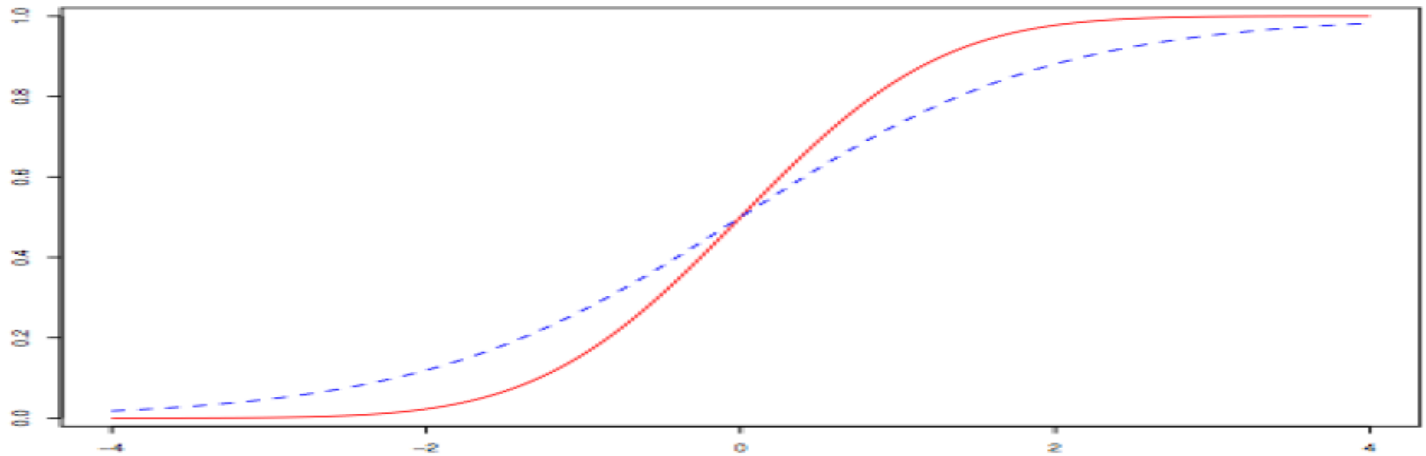


Figure: Fonctions de répartition de la fonction logistique (bleu) et probit (rouge).

Remarque

A partir de la remarque faite sur la représentation graphique, nous pouvons en déduire que

$$\mathbb{E}(y|\mathbf{X} = x) = f(x);$$

Où f est une fonction **sigmoïde**.

La remarque prouve que la probabilité de y_i notée π_i $\left(\pi_i = \mathbb{P}(Y_i = y_i|\mathbf{X}_i) = \mathbb{E}(Y_i|\mathbf{X}_i)\right)$ dépend explicitement des variables explicatives $\mathbf{X}_i = x_i$.

Questions

- Le choix d'un modèle linéaire de la forme

$$\pi_i = \mathbb{E}(Y_i|\mathbf{X}_i) = X_i^T\beta = \beta_1 X_{i,1} + \beta_2 X_{i,2} + \cdots + \beta_N X_{i,N}$$

  convient t'il?
- Quels types de modèles peuvent être envisagés?

La réponse à la première question est non car

- la probabilité $\pi_i \in [0,1]$ et aucune propriété ne garantit que $\mathbf{X}_i^T\beta \in [0,1]$.
- une fonction sigmoide n'est pas linéaire.

Supposons que nous possédons N observations $(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \cdots, (\mathbf{X}_N, y_N)$ avec

- La variable $\mathbf{X}_i \in \mathbb{R}^p$ est un vecteur de variables explicatives (covariables)
- La variable $y_i \in \{0, 1\}$ est la réponse binaire qui détermine le groupe de l'observation.

Objectif

Construire un modèle de classification binaire qui va prédire les classes des nouvelles observations.

En réalité, les variables $\mathbf{X}_i$ sont **déterministes** et les variables y_i sont **aléatoires**.

Les variables y_i suivent une loi de Bernoulli de paramètres π_i. On rappelle que

$$\pi_i := \mathbb{P}(Y_i = y_i | \mathbf{X}_i = x_i)$$

La fonction **logit** est définie sur $]0,1[$ par

$$\forall p \in]0,1[, \quad \textbf{logit}(p) = \log(\frac{p}{1-p}).$$

C'est une fonction dérivable et bijective sur $]0,1[$ vers $\mathbb{R}$.

L'image de la probabilité π_i :

$$\textbf{logit}(\pi_i) = \log\left(\frac{\mathbb{P}(Y_i = y_i|\mathbf{X}_i = x_i)}{1-\mathbb{P}(Y_i = y_i|\mathbf{X}_i = x_i)}\right) = x_i^T\beta;$$

avec $\beta \in \mathbb{R}^p$.

On obtient

$$\mathbb{P}(Y_i = y_i|\mathbf{X}_i = x_i) = \frac{\exp(x_i^T\beta)}{1+\exp(x_i^T\beta)}$$

Si $y_i = 1$ alors

$$\mathbb{P}(Y_i = 0|\mathbf{X}_i = x_i) = \frac{1}{1+\exp(x_i^T\beta)}$$

D'autres fonctions **sigmoïde** peuvent être utilisées à la place de la fonction **logit** :

- La fonction **probit** :

$$\forall p \in [0,1], \quad \textbf{probit}(p) = \phi^{-1}(p);$$

où ϕ est la fonction de distribution de la loi normale centrée réduite définie par

$$\phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{u} \exp\Big(-\frac{1}{2}t^2\Big)dt$$

- La fonction **log-log** :

$$\forall p \in]0,1[, \quad \textbf{log-log} = \log\Big(-\log(1-p)\Big).$$

En pratique, la fonction **logit** est largement utilisée à cause l'interprétation facile du paramètre β dans cette fonction.

Considérons N observations indépendantes et identiquement distribuées de réalisations $(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$. La fonction **vraisemblance** :

$$\begin{aligned} L_N(\beta) &= \prod_{i=1}^{N} \mathbb{P}(y_i = 1|x_i)^{y_i} (1 - \mathbb{P}(y_i = 1|x_i))^{1-y_i} \\ &= \prod_{i=1}^{N} \frac{\exp(y_i \beta^T x_i)}{1 + \exp(\beta^T x_i))} \end{aligned}$$

La fonction **log-vraisemblance** :

$$\mathcal{L}_N(\beta) = \sum_{i=1}^{N} \left(y_i \beta^T x_i - \log(1 + e^{\beta^T x_i}) \right)$$

Le problème d'optimisation (maximiser la **log-vraisemblance**) :

$$\hat{\beta}_N = arg \max_{\beta} \mathcal{L}_N(\beta).$$

La dérivée partielle par rapport à la variable β_j ($j \in \{1, 2, \cdots, p\}$) :

$$\begin{aligned}\frac{\partial \mathcal{L}_N}{\partial \beta_j} &= \sum_{i=1}^{N} \left(y_i x_{i,j} - x_{i,j} \cdot \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \right) \\ &= \sum_{i=1}^{N} x_{i,j} \left(y_i - \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \right), \quad \forall j \in \{1, 2, \cdots, p\}\end{aligned}$$

La dérivée partielle par rapport à β_j peut s'écrire sous la forme matricielle :

$$\mathcal{T}_N(\beta) = \frac{\partial \mathcal{L}_N}{\partial \beta_j} = \sum_{i=1}^{N} x_{i,j} \left(y_i - \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \right)$$

L'estimateur $\hat{\beta}$ est solution du système p équations.

$$\begin{cases} \sum_{i=1}^{N} x_{i,1} \left(y_i - \frac{e^{y_i \beta^T x_i}}{1+e^{y_i \beta^T x_i}} \right) = 0 \\ \vdots \quad \vdots \quad \vdots \\ \sum_{i=1}^{N} x_{i,p} \left(y_i - \frac{e^{y_i \beta^T x_i}}{1+e^{y_i \beta^T x_i}} \right) = 0 \end{cases} \tag{4}$$

La solution exacte du système d'équations n'existe pas. Les estimations de $\hat{\beta}$ se font avec l'algorithme numérique de **Newton-Raphson**.

- Sous certaine condition de **séparabilité**, la fonction **log-vraisemblance** est **concave** et la méthode du maximum de vraisemblance converge vers un unique maximum.
- Le choix du point de départ pour l'algorithme numérique n'est pas critique. On peut commencer par 0 ou par un point aléatoire.

Algorithme de Newton-Raphson : une méthode numérique qui permet de déterminer la racine d'une fonction mathématique $F(\beta)$.
Dans notre cas, on pose

$$F(\beta) = \Big(\sum_{i=1}^{N} x_{i,1} \Big(y_i - \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \Big), \sum_{i=1}^{N} x_{i,2} \Big(y_i - \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \Big), \cdots, \sum_{i=1}^{N} x_{i,p} \Big(y_i - \frac{e^{y_i \beta^T x_i}}{1 + e^{y_i \beta^T x_i}} \Big) \Big)$$

Algorithme de Newton-Raphson

1. Initialisation : on donne $\beta^{(0)}$
2. Approximation linéaire de la fonction F au point initial $\beta^{(0)} + h$:

$$F(\beta^{(0)} + h) \simeq F(\beta^{(0)}) + hF'(\beta^{(0)})$$

3. Déterminer une solution $\beta^{(1)} = \beta^{(0)} + h$ telle que $F(\beta^{(1)}) = 0$ implique $h = -[F'(\beta^{(0)})]^{-1}F(\beta^{(0)})$. Donc

$$\beta^{(1)} = \beta^{(0)} - [F'(\beta^{(0)})]^{-1}F(\beta^{(0)})$$

4. Itérer le processus jusqu'à ce que le critère de convergence soit satisfait

Dans le cas du modèle logistique, l'algorithme de Newton-Raphson porte sur la résolution du système

$$F(\beta) = \frac{\partial \mathcal{L}_N}{\partial \beta} = 0_{\mathbb{R}^p}.$$

1. Initier $\beta^{(0)}$
2. Pour tout $k \geq 0$, calculer

$$\beta^{(k+1)} = \beta^{(k)} - \left(\frac{\partial^2 \mathcal{L}_N}{\partial \beta \partial \beta^T} \Big|_{\beta^{(k)}} \right)^{-1} \frac{\partial \mathcal{L}_N}{\partial \beta} |_{\beta^{(k)}}$$

3. Itérer le processus jusqu'à ce que $\beta^{(k+1)} \approx \beta^{(k)}$ et/ou $\mathcal{L}_N(\beta^{(k+1)}) \approx \mathcal{L}_N(\beta^{(k)})$.

Posons que $\mathbf{X}$ la matrice des covariables de N lignes (nombres d'observations) et p colonnes (nombre de variables explicatives) :

$$\mathbf{X} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,p} \end{pmatrix}$$

Posons que $y = \left(y_1, y_2, \cdots, y_N \right)^T$ et

$\Phi(\beta) = (\phi(\beta^T x_1), \phi(\beta^T x_2), \cdots, \phi(\beta^T x_N))$ avec $\phi(u) = \frac{e^u}{1+e^u}, \forall u \in \mathbb{R}$.

Le système d'équations (4) a la forme suivante

$$\mathbf{X}^T\Big(y - \Phi(\beta)\Big) = 0.$$

L'élément de la ligne j et de la colonne k de la matrice Hessienne :

$$\begin{aligned}
\frac{\partial \mathcal{L}_N}{\partial \beta_j \partial \beta_k} &= -\sum_{i=1}^{N} x_{i,j} x_{i,k} \frac{e^{y_i \beta^T x_i}}{(1 + e^{y_i \beta^T x_i})^2} \\
&= -\sum_{i=1}^{N} x_{i,j} x_{i,k} \phi(\beta^T x_N)(1 - \phi(\beta^T x_N)) \\
&= -\sum_{i=1}^{N} x_{i,j} \phi(\beta^T x_N)(1 - \phi(\beta^T x_N)) x_{i,k}
\end{aligned}$$

Alors

$$\frac{\partial^2 \mathcal{L}_N}{\partial \beta \partial \beta^T} = -\mathbf{X}^T \mathbf{W}(\beta) \mathbf{X},$$

avec

$$\mathbf{W}(\beta) = diag\Big((\phi(\beta^T x_1)(1 - \phi(\beta^T x_1)), \cdots, \phi(\beta^T x_N)(1 - \phi(\beta^T x_N)))^T\Big)$$

En utilisant les écritures matricielles

$$\frac{\partial \mathcal{L}_N}{\partial \beta} = \mathbf{X}^T\Big(y - \Phi(\beta)\Big), \quad \frac{\partial^2 \mathcal{L}_N}{\partial \beta \partial \beta^T} = -\mathbf{X}^T\mathbf{W}(\beta)\mathbf{X}.$$

L'algorithme de Newton-Raphson devient :

$$\begin{aligned}
\beta^{(k+1)} &= \beta^{(k)} - \Big(\mathbf{X}^T\mathbf{W}(\beta^{(k)})\mathbf{X}\Big)^{-1}\mathbf{X}^T\Big(y - \Phi(\beta^{(k)})\Big) \\
&= \Big(\mathbf{X}^T\mathbf{W}(\beta^{(k)})\mathbf{X}\Big)^{-1}\mathbf{X}^T\mathbf{W}(\beta^{(k)}) \\
&\quad \Big(\mathbf{X}\beta^{(k)} - \mathbf{W}^{-1}(\beta^{(k)})\Big(y - \Phi(\beta^{(k)})\Big)\Big) \\
&= \Big(\mathbf{X}^T\mathbf{W}(\beta^{(k)})\mathbf{X}\Big)^{-1}\mathbf{X}^T\mathbf{W}(\beta^{(k)})\mathbf{Z};
\end{aligned}$$

Où $\mathbf{Z} = \Big(\mathbf{X}\beta^{(k)} - \mathbf{W}^{-1}(\beta^{(k)})\Big(y - \Phi(\beta^{(k)})\Big)\Big)$.

Problème de convergence :

Le problème de convergence de l'estimateur du maximum de vraisemblance peut être lié à la séparabilité des classes.

Définition

Un nuage de points $(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$ avec $x_i \in \mathbb{R}^p$ et $y_i \in \{0, 1\}$, est dit

- **complètement séparable** si $\exists \beta \in \mathbb{R}^p$ tel que $\forall i,\ y_i = 1$ on a $\beta^T x_i > 0$ et $\forall i,\ y_i = 0$ on a $\beta^T x_i < 0$.
- **quasi-complètement séparable** si $\exists \beta \in \mathbb{R}^p$ tel que $\forall i,\ y_i = 1$ on a $\beta^T x_i \geq 0$, $\forall i,\ y_i = 0$ on a $\beta^T x_i \leq 0$ et $\{i : \beta^T x_i = 0\} \neq \emptyset$.
- **en recouvrement** ("overlap data") s'il n'est ni complètement séparable et ni quasi-complètement séparable.

L'estimateur du maximum de vraisemblance ne converge pas si les données sont complètement séparées et quasi-complètement séparées.

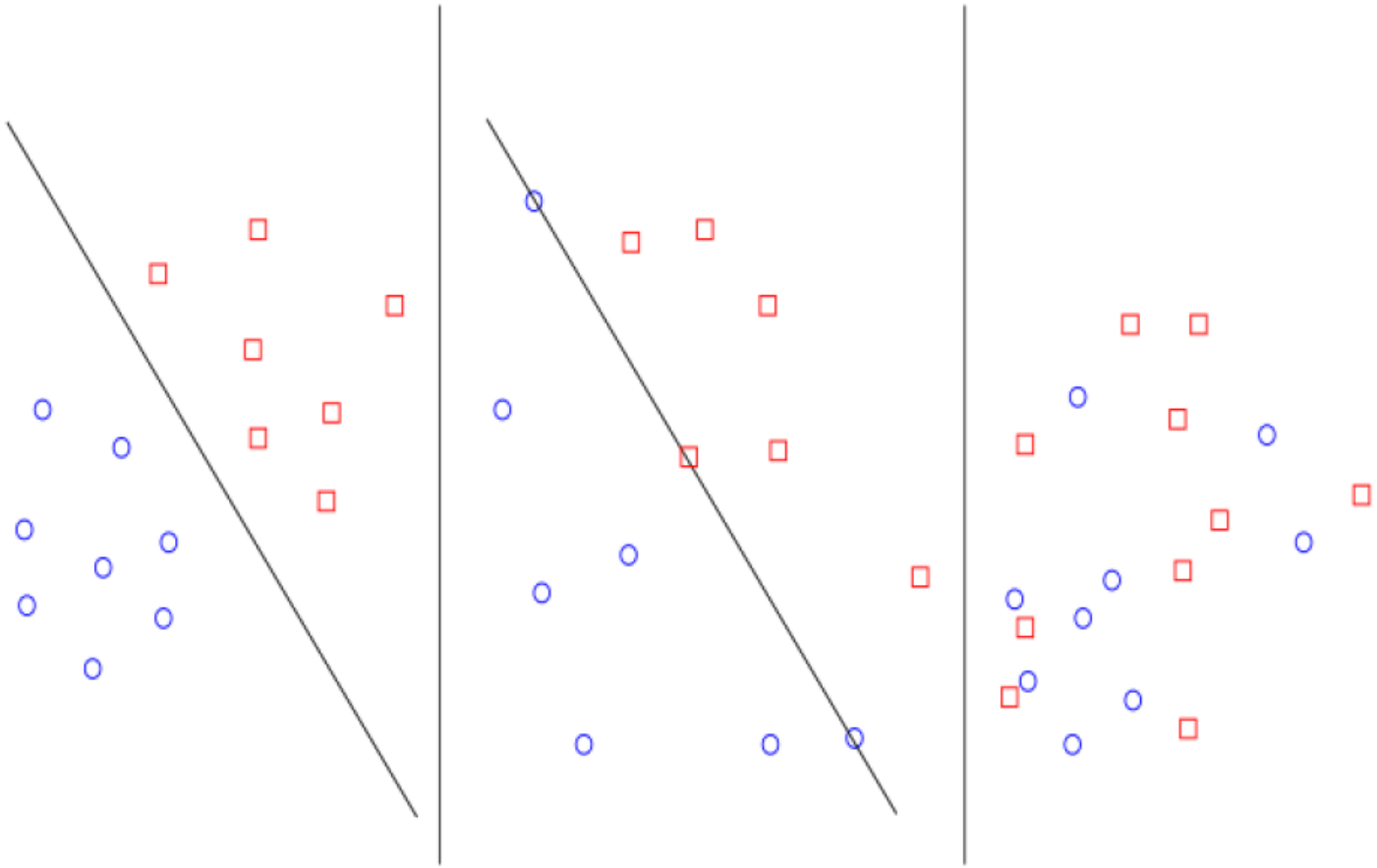


Figure: A gauche : données complètement séparables. Au milieu : données quasi-complètement séparables. A droite : données en recouvrement (**overlap data**).

Théorème

Sous hypothèses des données en recouvrement, l'estimateur du maximum de vraisemblance $\hat{\beta}$ est consistant et $\sqrt{n}\left(\hat{\beta}-\beta\right)_{n\in\mathbb{N}^*}$ converge en loi vers $\mathcal{N}\left(0,\mathcal{I}(\beta)^{-1}\right)$ où $\mathcal{I}(\beta)$ est la matrice d'information de Fisher définie par

$$\mathcal{I}(\beta)_{i,j} = -\mathbb{E}\left(\frac{\partial^2}{\partial\beta_i\partial\beta_j}L(\beta)\right),$$

avec $L(\beta)$ est la **log-vraisemblance** d'une observation.

L'estimation de $\mathcal{I}(\beta)$ est nécessaire pour calculer les intervalles de confiance pour β et pour tester des hypothèses sur β.

- Soit $L_{(k)}(\beta)$ la contribution de l'observation k dans la log-vraisemblance $\mathcal{L}_N(\beta)$. C'est-à-dire

$$\mathcal{L}_N(\beta) = \sum_{k=1}^{N} L_{(k)}(\beta).$$

- La matrice inconnue $\mathcal{I}(\beta)$ est estimée

$$\begin{aligned}\hat{\mathcal{I}}(\beta) &= -\frac{1}{N}\sum_{k=1}^{N}\frac{\partial^2}{\partial\beta\partial\beta^T}L_{(k)}(\beta)\Big) = -\frac{1}{N}\frac{\partial^2}{\partial\beta\partial\beta^T}\sum_{k=1}^{N}L_{(k)}(\beta)\Big) \\ &= -\frac{1}{N}\frac{\partial^2}{\partial\beta\partial\beta^T}\mathcal{L}_N(\beta) = \mathbf{X}^T\mathbf{W}(\beta)\mathbf{X}\end{aligned}$$

- Puisque les paramètres β sont inconnus alors on calcule

$$\hat{\mathcal{I}}(\hat{\beta}) = \mathbf{X}^T\mathbf{W}(\hat{\beta})\mathbf{X},$$

où $\hat{\beta}$ est calculé avec l'algorithme de Newton-Raphson.

Sommaire

En pratique,

- différents modèles peuvent se présenter en fonction des nombres de covariables (variables explicatives)
- le choix du meilleur modèle est une étape cruciale en machine learning

Considérons n modèles $\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_n$.

Question

Comment choisir le **meilleur modèle** parmis ces modèles?

- Il n'existe pas de **critère universel** de définition du **meilleur modèle**.
- Le **meilleur modèle** dépend d'un **critère donné**.

Plusieurs types de critères de selection du meille:

- Tests sur les paramètres des modèles emboités
- **Critère d'information d'Akaike** : AIC
- **Critère d'information bayésien** : BIC

Tests sur les paramètres

Considérons deux modèles $\mathcal{M}_1$ et $\mathcal{M}_2$.
On suppose que le modèle $\mathcal{M}_1$ est emboité dans le modèle $\mathcal{M}_2$ ($\mathcal{M}_1$ est un cas particulier de $\mathcal{M}_2$).
Posons que

$$\mathcal{M}_1 : \textbf{logit}(\pi_i) = \beta_1 x_1 + \beta_2 x_2;$$
$$\mathcal{M}_2 : \textbf{logit}(\pi_i) = \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4.$$

Le test de comparaison des modèles $\mathcal{M}_1$ et $\mathcal{M}_2$:

$$H_0 : \beta_3 = \beta_4 = 0 \quad \text{contre} \quad H_0 : \beta_3 \neq 0, \beta_4 \neq 0$$

En général, les deux modèles contiennent respectivement p_1 et p_2 paramètres et l'un des deux modèles est emboîté dans l'autre.

- Le test de comparaison des deux modèles porte sur la nullité de certains paramètres : **Wald** et **du rapport de vraisemblance**
- Sous l'hypothèse H_0, les statistiques suivent une loi de **chi2** de degré de liberté $p_2 - p_1$ si $p_2 > p_1$

Critère d'information d'Akaike (AIC)

Soit un modèle $\mathcal{M}$ de p paramètres estimés par l'estimateur de maximum de vraisemblance $\hat{\beta}$.
Le critère **AIC** est une méthode de pénalisation de la log-vraisemblance :

$$\mathbf{AIC}(\mathcal{M}) = -2\mathcal{L}_N(\hat{\beta}) + 2p$$

Idée

Il faut choisir le modèle qui a la plus grande log-vraisemblance sachant que la log-vraisemblance croît en fonction la complexité du modèle (le nombre de paramètres).
Intuitivement, le modèle ayant la plus grande log-vraisemblance est le modèle complet mais à retenir que ce modèle est sur-paramétré (appelé "overfitting").

Le critère **AIC** permet de pénaliser les modèles avec le nombre de paramètres afin de satisfaire des critères.

Critère d'information bayésien (BIC)

Soit un modèle $\mathcal{M}$ de p paramètres estimés par l'estimateur de maximum de vraisemblance $\hat{\beta}$.
Le critère **BIC** est inspirée du critère **AIC**. Pour un échantillon de N observations. Le critère **BIC** est défini par

$$\mathbf{BIC}(\mathcal{M}) = -2\mathcal{L}_N(\hat{\beta}) + p\log(N)$$

Idée

Choisir un modèle dont les valeurs de **AIC** et **BIC** sont petites.
Si $\log(N) > 2$ ($N > 8$), le critère **BIC** aura tendance à choisir le modèle le plus parcimonieux que le critère **AIC**.

La validation d'un modèle est basée sur le pouvoir de prédiction et se fait en plusieurs étapes :

- Pour chaque modèle, déterminer le nombre d'observations mal classées
- Calculer les taux d'erreur des modèles

L'approche consiste à définir une **règle de classification** des observation à partir d'un modèle logistique :

$$\begin{array}{rcl} \mathcal{G} & : & \mathbb{R}^p \to \{0,1\} \\ & & \mathbf{X} \mapsto y \end{array}$$

Le modèle logistique :

$$\mathbb{P}(y = y_i | \mathbf{X} = x) = \frac{\exp(\hat{\beta}^T x_i)}{1 + \exp(\hat{\beta}^T x_i)}$$

Pour une nouvelle observation $\mathbf{X}_{N+1}$, on

$$\mathcal{G}(\mathbf{X}_{N+1}) = \begin{cases} y_i & \text{si} \quad \mathbb{P}(y = y_i | \mathbf{X} = x_{N+1}) \geq s \\ 1 - y_i & \text{sinon} \end{cases} \tag{5}$$

s est le seuil fixé.

Il existe plusieurs critères de mesure de performance d'une **règle de classification** dont l'estimation de la **probabilité d'erreur** $\mathbb{P}(\mathcal{G}(\mathbf{X}) \neq y)$.

Soit $(\mathbf{X}_i)$ une suite d'observations prédites dans les classes $\mathcal{G}(\mathbf{X}_i)$. La proportion des observations mal classées :

$$P_{ml}(\mathcal{G}) = \frac{1}{N} \sum_{i=1}^{N} 1_{\mathcal{G}(\mathbf{x}_i) \neq y_i}$$

Un modèle qui classe bien toutes les observations (modèle parfait) a une proportion des mal classées égale à 0.

Problèmes

- $P_{ml}(\mathcal{G})$ n'est pas un **bon estimateur** de la probabilté $\mathbb{P}(\mathcal{G}(\mathbf{X}) \neq y)$.
- La théorie des grands nombres ne peut pas être appliquée car les $1_{\mathcal{G}(\mathbf{x}_i) \neq y_i}$ ne sont pas indépendantes.
- La base de données **train set** est utilisée deux fois pour calculer $\mathcal{G}$ et $P_{ml}(\mathcal{G})$.

Solution

Dans le cas d'une base de données riche et bien traitée, composée des éléments $(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$, on partitionne aléatoirement l'échantillon en deux parties :

- un **échantillon d'entrainement** (**train set**) pour estimer la fonction $\mathcal{G}$ de taille q, noté $\mathcal{A}_q = \{(x_i, y_i), i \in E_q\}$
- un **échantillon de test ou de validation** (**test set**) pour estimer la probabilité $P_{ml}(\mathcal{G})$ de taille $N - q$, noté $\mathcal{V}_{N-q} = \{(x_i, y_i), i \in E_{N-q}\}$.

$$\hat{P}_{ml}(\mathcal{G}) = \frac{1}{N-q} \sum_{i \in E_{N-q}} 1_{\mathcal{G}(\mathbf{x}_i) \neq y_i},$$

$$E_q \cup E_{N-q} = \{1, 2, \cdots, N\}, \; E_q \cap E_{N-q} = \emptyset$$

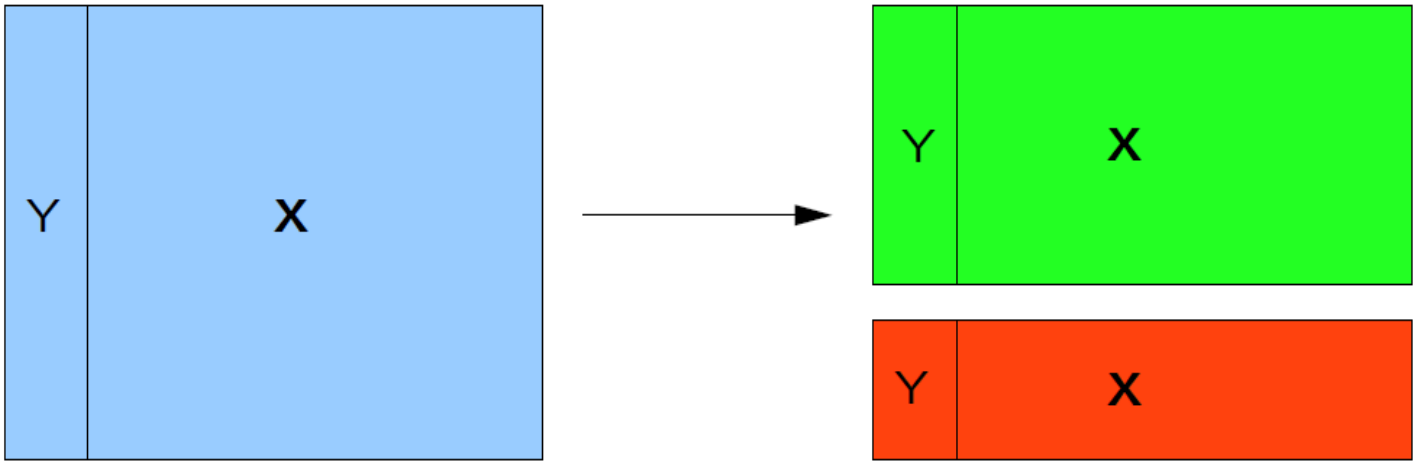


Figure: Base initiale (bleue), **train set**(vert) et **test set**(rouge).

- $\hat{P}_{ml}(\mathcal{G})$ est un estimateur sans biais de $P_{ml}(\mathcal{G})$
- On retient le modèle ayant la plus petite valeur de $\hat{P}_{ml}(\mathcal{G})$

Remarque : Il est difficile de donner une règle générale sur la manière de choisir le nombre d'observations dans les bases de données d'entrainement et de test car cela dépend du rapport signal/bruit dans les données et de la complexité des modèles.

Validation croisée (Cross-validation)

Quelques inconvénients de la procédure basée sur la partition **train/test**

- Il faut une grande base de données pour une correcte estimation des paramètres avec **train set** et une meilleure évaluation des erreurs sur le **test set**.
- Les résultats de la procédure dépendent de la composition des bases de données **train/test** et **train set**.

Pour surmonter ces difficultés, la méthode de validation croisée (cross-validation) peut être envisagée.

- La méthode la plus simple et la plus utilisée pour faire de la prédiction des erreurs.
- Lorsqu'il y'a largement de données, il est possible de retirer des données qui sont utilisées pour la validation. Cela n'est pas possible lorsqu'il y'a moins de données.

Validation croisée en $K - blocs$

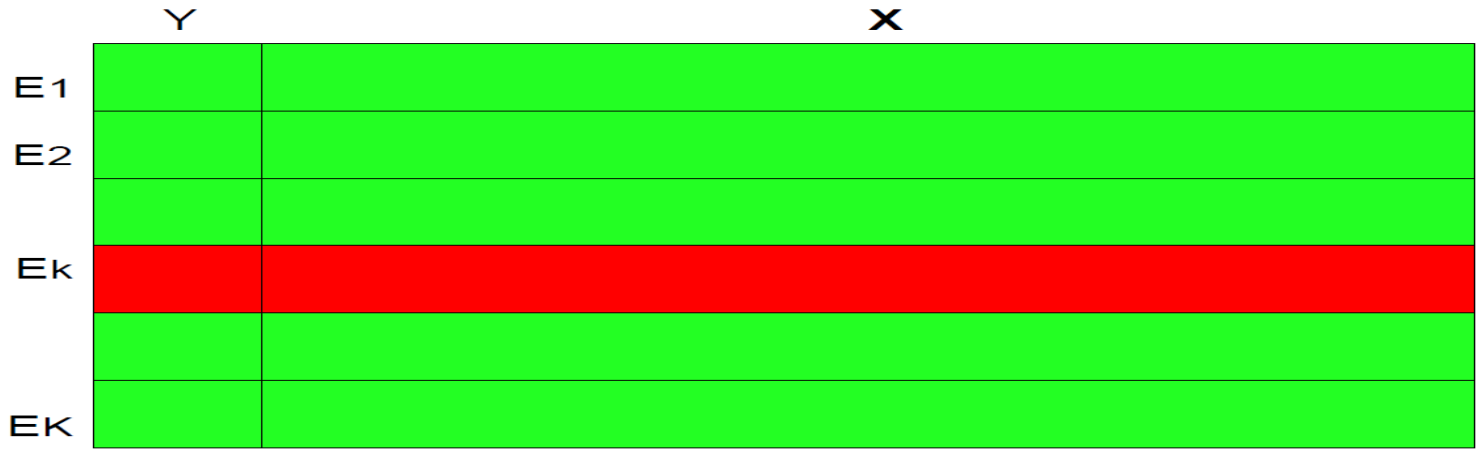
Subdiviser la base de données en K sous-échantillons E_k de même taille ($k \in \{1, 2, \cdots, K\}$. Cela donne K **train/test** procédures à mener.
Pour la procédure d'ordre k :

- L'échantillon **train set** : utilisé pour estimer $\hat{\beta}$
- L'échantillon **test set** : utilisé pour estimer l'erreur de prédiction.

Dans la procédure d'ordre k, nous obtenons une prédiction des classes y pour chaque échantillon E_k.
A la fin de la procédure, une prédiction de y est disponible pour chacune des observations de la base de données initiale. Ces prédictions sont utilisées pour calculer la prédiction erreur. Nous trouvons le modèle avec la plus petite erreur.
Soit $\kappa : \{1, 2, \cdots, N\} \rightarrow \{1, 2, \cdots, K\}$ la fonction indiquant la partition aléatoire dans laquelle se trouve l'observation i de la base de données.
Soit $\hat{y}_i^{\kappa(i)}$ la prédiction de y_i dans l'échantillon $\kappa(i)$ retiré des autres données.

$$CV_{\kappa(i)} = \frac{1}{N} \sum_{i=1}^{card(E_{\kappa(i)})} 1_{\hat{y}_i^{\kappa(i)} \neq y_i}.$$

Le meilleur modèle est celui avec la petite valeur de $CV_{\kappa(i)}$.



Le meilleur choix du paramètre K.

- Si K est petit, le nombre de données dans les **train set** est petit,
- Si K est grand, le nombre de données dans les **test set** est petit

Les valeurs typiques de K:

- $K = 2$: Validation croisée à 2 blocs. Deux sous-échantillons de même taille sont utilisés pour **train/test set**.
- $K = 5$
- $K = 10$
- $K = N$: **leave-one-out** cross-validation (LOOCV)
  Les échantillons **train set** et **test set** contiennent respectivement $N - 1$ observations et une observation.

Dans l'ensemble, les validations croisées à 5 ou 10 blocs sont généralement recommandées comme un bon compromis.

Matrice de confusion

Erreurs de pédiction : il existe deux types d'erreurs de prédictions

- une observation $y = 0$ peut être prédite $\hat{y} = 1$
- une observation $y = 1$ peut être prédite $\hat{y} = 0$.

Il est souvent intéressant de déterminer le type d'erreur commise.
La **matrice de confusion** est un moyen pratique pour afficher les informations concernant les erreurs.

	$\hat{y}$=0	$\hat{y}$=1	Total
y=0	Vraie Négative (TN)	Fausse Positive (FP)	Négative (N)
y=1	Fausse Négative (FN)	Vraie Positive (TP)	Positive (P)
	$\hat{N}$	$\hat{P}$	

A partir de la **matrice de confusion**, on définit les mesures de performance suivantes

- Précision (**precision**) : taux de positifs parmis les positifs prédits et utile lorsque les FP ont des conséquences graves.

$$\textbf{precision} = \frac{TP}{\hat{P}} = \frac{TP}{TP + FP}.$$

- Rappel ou Sensibilité (**sensitivity** or **recall**) : taux de positifs parmis les positifs observés.

$$\textbf{recall} = \textbf{sensivility} = \frac{TP}{P} = \frac{TP}{TP + FN}.$$

- Spécifité (**specificity**) : taux de négatifs prédits parmis les négatifs observés et utile lorsque les FN ont des conséquences graves.

$$\textbf{specificity} = \frac{TN}{N} = \frac{TN}{TN + FP}.$$

- **F1–Score** : utile lorsque les deux classes ne sont pas équilibrées.

$$\textbf{F1–Score} = \frac{\textbf{2*Precision*Recall}}{\textbf{Precision+Recall}}.$$

- Exactitude (**accuracy score**) : la proportion de prédictions correctes parmi le nombre total de cas examinés.

$$\textbf{accuracy score} = \frac{TP + TN}{N + P} = \frac{TP + TN}{TP + TN + FP + FN}.$$

Les inconveniants des mesures de performance

- **precision** et **recall** peuvent être trompeurs si les deux classes ne sont pas équilibrées.
- **F1–Score** peut-être biaisé si l'une des valeurs (la précision ou le rappel) est plus importante que l'autre.

Ces mesures de performances dépendent de la valeur du seuil s donnée dans la formule (5).

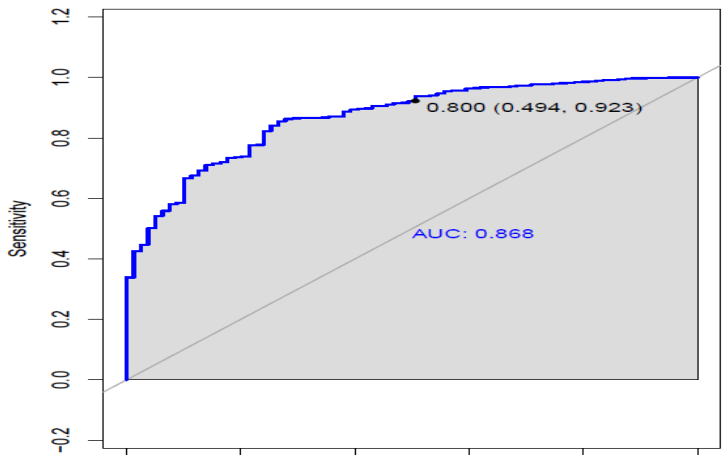
- Quand s augmente, **sensivility** ou **recall** décroit et **specificity** augmente.
- Un bon modèle est celui qui donne les grandes valeurs de **sensivility** et de **specificity**.

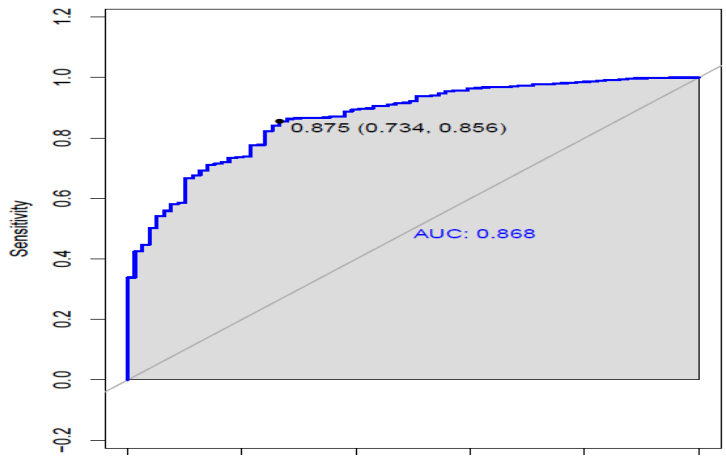
Courbe ROC et AUC

Courbe ROC

La courbe **Receiver Operating Characteristic (ROC)** est une courbe paramétrée de paramètre le seuil s. Elle représente les évolutions de **sensivility** et de **1 -specificity** en faisant varier le seuil s.

- Les courbes **ROC** sont utiles pour comparer différents modèles puisqu'ils prennent en compte tous les seuils possibles.
- La performance globale de classification du modèle dans l'ensemble des seuils possibles sont résumés par la zone sous la courbe **ROC** (**AUC**).





- Pour un classifier non aléatoire, la courbe **ROC** est au dessus de la ligne diagonale (**AUC**>0.5)
- Le meilleur classifier est celui qui a la plus grande valeur de **AUC**
- L'aire entre la courbe **ROC** et la ligne diagonale est égale **AUC** $-$ 0.5 et **Gini coefficient** $= 2$**AUC** $- 1$
- Plusieurs modèles peuvent être comparés en superposant les courbes **ROC** sur le même graphe.