Troisième partie

Apprentissage supervisé

Chapitre 7 Minimisation du risque empirique

Notions : classification, régression, espace des hypothèses, minimisation du risque empirique, moindres carrés, modèles paramétriques linéaires

Objectifs pédagogiques :

- Formaliser un problème d'apprentissage supervisé.
- Décrire l'espace des hypothèses dans le cas d'un modèle paramétrique.
- ★ Prouver l'équivalence entre maximisation de la vraisemblance et minimisation du risque empirique dans le cas gaussien.
- Mettre en œuvre une régression linéaire.

Nous nous intéressons maintenant aux problèmes d'apprentissage **supervisé** : il s'agit de développer des algorithmes qui soient capables d'apprendre des modèles **prédictifs**. À partir d'exemples étiquetés, ces modèles seront capables de prédire l'étiquette de nouveaux objets. Le but de ce chapitre est de développer les concepts généraux qui nous permettent de formaliser ce type de problèmes.

7.1 Formalisation d'un problème d'apprentissage supervisé

Nous supposons maintenant disposer non seulement d'une matrice $X \in \mathbb{R}^{n\times p}$ décrivant n individus en p dimensions, mais aussi de n **étiquettes** $\{y^1,y^2,\dots,y^n\}$. Chaque étiquette y^i appartient à un espace $\mathcal{Y}$. Dans ce cours, nous allons considérer deuxcas particuliers pour $\mathcal{Y}$:

- $\mathcal{Y} = \mathbb{R}$: on parle d'un problème de **régression** ;
- $\mathcal{Y} = \{0,1\}$: on parle d'un problème de **classification binaire**, et les observations dont l'étiquette vaut 0 sont appelées **négatives** tandis que celles dont l'étiquette vaut 1 sont appelées **positives**. Dans certains cas, il sera mathématiquement plus simple d'utiliser $\mathcal{Y} = \{-1,1\}$.

La matrice $X \in \mathbb{R}^{n\times p}$ telle que $X_{ij} = x_j^i$ soit la j-ème variable du i-ème individu est appelée **matrice de données** ou **matrice de design**.

On peut aussi choisir de représenter chaque individu et son étiquette par le couple $(\vec{x}^{\,i},y^i) \in \mathbb{R}^p\times\mathcal{Y}$. L'ensemble $\mathcal{D} = \{(\vec{x}^{\,i},y^i)\}_{i=1,\dots,n}$ forme alors le **jeu d'apprentissage**.

Le machine learning étant issu de plusieurs disciplines et champs d'applications, on trouvera plusieurs noms pour les mêmes objets. Ainsi les variables sont aussi appelées **descripteurs**, **attributs**,

prédicteurs, ou **caractéristiques** (en anglais, *variables, descriptors, attributes, predictors* ou encore *features*). Les **individus**, ou **observations** sont aussi appelées **exemples**, **échantillons** ou **points du jeu de données** (en anglais, *samples* ou *data points*). Enfin, les étiquettes sont aussi appelées **variables cibles** (en anglais, *labels, targets* ou *outcomes*).

Le but de l'apprentissage supervisé est alors de trouver une fonction $f : \mathbb{R}^p \to \mathcal{Y}$ telle que $f(\vec{x}) \approx y$, qui s'applique non seulement aux n individus observés, mais plus généralement à tous les individus d'une population à laquelle on suppose que ces n individus appartiennent. C'est cette fonction f qui est le **modèle prédictif** appris. Un **algorithme d'apprentissage supervisé** utilise le jeu de données $\mathcal{D}$ données pour déterminer f.

Plus formellement, supposons que les couples $(\vec{x}^{\,i}, y^i)$ soient les réalisations de n vecteurs aléatoires de même loi qu'un couple de variables aléatoire (X,Y), X étant un vecteur aléatoire p-dimensionnel et Y une variable aléatoire réelle à valeurs dans $\mathcal{Y}$. Supposons de plus qu'il existe une fonction $\Phi : \mathbb{R}^p \to \mathcal{Y}$ et une variable aléatoire réelle ϵ telle que

$$Y = \Phi(X) + \epsilon, \tag{7.1}$$

ϵ représentant un **bruit**. Ce bruit peut être causé

- par des *erreurs de mesure* dues à la faillibilité des capteurs utilisés pour mesurer les variables par lesquelles on représente nos données, ou à la faillibilité des opérateurs humains qui ont entré ces mesures dans une base de données ;
- par des *erreurs d'étiquetage* (souvent appelés *teacher's noise* en anglais) dues à la faillibilité des opérateurs humains qui ont étiqueté les données ;
- enfin, parce que les variables mesurées ne suffisent pas à modéliser le phénomène qui nous intéresse, soit qu'on ne les connaisse pas, soit qu'elles soient coûteuses à mesurer.

Notre but est d'approcher Φ par f.

Dans le cas d'un problème de classification, le modèle prédictif peut prendre directement la forme d'une fonction f à valeurs dans $\{0,1\}$, ou utiliser une fonction intermédiaire g à valeurs réelles, qui associe à une observation un score d'autant plus élevé qu'elle est susceptible d'être positive. Ce score peut par exemple être la probabilité que cette observation appartienne à la classe positive. On obtient alors f en **seuillant** g ; g est appelée **fonction de décision** [1].

Exemple

Filtrage de spam. On peut poser le filtrage de spam comme un problème de classification binaire. Les individus sont des emails. Leur étiquette est binaire (positive pour « spam » et négative pour « non-spam »). Les p variables représentant un email peuvent être définies comme le nombre d'occurrences, pour p mots, de chacun de ces mots dans l'email (p est ainsi la taille d'un dictionnaire pré-défini) [2]. Étant donné un jeu de données de n emails étiquetés, un algorithme d'apprentissage retourne une fonction f qui, à tout email représenté par un vecteur de $\mathbb{R}^p$ (en fait, $\mathbb{N}^p$), associe une étiquette 0 ou 1. Ce modèle $f : \mathbb{R}^p \to \{0,1\}$ peut être obtenu en seuillant une fonction de décision $g : \mathbb{R}^p \to \mathbb{R}$.

Le bruit peut être dû aux causes suivantes :

- Des erreurs de mesures peuvent être causées par des fautes d'orthographe (volontaires ou non) qui empêchent de comptabiliser certains mots.
- Des erreurs d'étiquetage peuvent arriver quand une personne marque par erreur comme

1. Dans la librairie `scikit-learn`, on fera ainsi attention à la distinction entre les méthodes `predict` et `predict_proba`.
2. C'est ce qu'on appelle une représentation *bag-of-words*

courrier indésirable un email qui ne l'était pas, ou, inversement, laisse dans sa boîte mail ou supprime sans étiqueter comme tel un email indésirable.

— Enfin, notre représentation est limitée, en particulier parce qu'elle ne prend pas en compte l'ordre des mots. Nous ne disposons pas de suffisamment d'information pour classifier les emails aussi efficacement qu'un humain.

Remarque. Les notions développées jusqu'à la fin de la section 7.4 peuvent l'être en remplaçant $\mathbb{R}^p$ par un espace quelconque $\mathcal{X}$.

7.2 Espace des hypothèses

Pour poser un problème d'apprentissage supervisé, il nous faut décider du type de modèles que nous allons considérer.

On appelle **espace des hypothèses** l'espace de fonctions $\mathcal{F}$, qui est un sous-espace de toutes les fonctions de $\mathbb{R}^p \to \mathcal{Y}$ décrivant les modèles que nous allons considérer. Cet espace est choisi en fonction de nos *convictions* par rapport au problème, ainsi que de considérations pratiques sur notre capacité à trouver facilement un « bon » modèle dans $\mathcal{F}$.

Le choix de l'espace des hypothèses est fondamental. En effet, si cet espace ne contient pas le « bon » modèle, il sera impossible de trouver une bonne fonction de décision. Cependant, si l'espace est trop générique, il sera plus difficile et intensif en temps de calcul d'y trouver une bonne fonction de modélisation.

Exemple

Dans l'exemple de la figure 7.1, on pourra décider de se restreindre à des discriminants qui soient des ellipses à axes parallèles aux axes de coordonnées. Ainsi, l'espace des hypothèses sera

$$\mathcal{F} = \{\vec{x} \mapsto \alpha(x_1 - a)^2 + \beta(x_2 - b)^2 - 1 \,;\, (\alpha,\beta,a,b) \in \mathbb{R}^4\}. \tag{7.2}$$

Dans cet espace, il semble possible de trouver un modèle f qui sépare les positifs des négatifs. Si nous avions choisi comme espace des hypothèses l'ensemble des fonctions linéaires de $\mathbb{R}^2$ dans $\mathbb{R}$, ce ne serait pas possible.

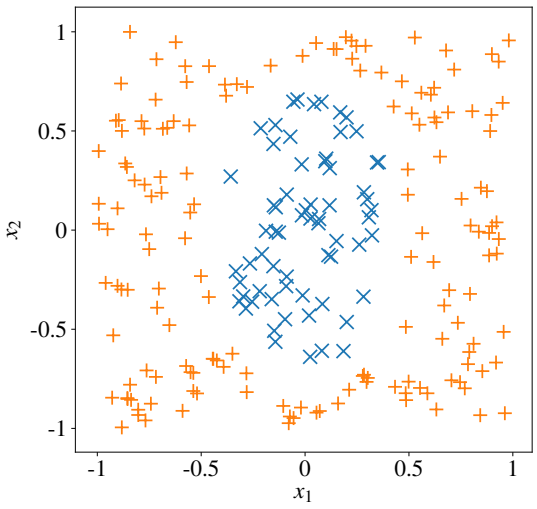


FIGURE 7.1 – Les exemples positifs (+) et négatifs (x) semblent être séparables par une ellipse.

La tâche d'apprentissage supervisé consiste à déterminer une hypothèse $f \in \mathcal{F}$ qui approche au mieux la fonction cible ϕ (voir équation (7.1)). Pour réaliser une telle tâche, nous allons développer dans les sections suivantes deux outils supplémentaires :

1. Une façon de **quantifier la qualité d'une hypothèse**, afin de pouvoir déterminer si une hypothèse satisfaisante (voire optimale) a été trouvée. Pour cela, nous allons définir la notion de **fonction de coût**.
2. Une façon de **chercher une hypothèse optimale** dans $\mathcal{F}$. Les algorithmes d'apprentissage supervisé que nous allons étudier ont pour but de trouver dans $\mathcal{F}$ l'hypothèse optimale au sens de la fonction de coût. Différents algorithmes correspondent à différents $\mathcal{F}$, et selon les cas cette recherche sera exacte ou approchée.

7.3 Minimisation du risque empirique

Résoudre un problème d'apprentissage supervisé revient à trouver une fonction $f \in \mathcal{F}$ dont les prédictions soient les plus proches possibles des véritables étiquettes, sur tout l'espace $\mathbb{R}^p$. On utilise pour formaliser cela la notion de **fonction de coût** :

Une **fonction de coût** $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, aussi appelée **fonction de perte** ou **fonction d'erreur** (en anglais : *cost function* ou *loss function*) est une fonction utilisée pour quantifier la qualité d'une prédiction : $L(y,f(\vec{x}))$ est d'autant plus grande que l'étiquette $f(\vec{x})$ est éloignée de la vraie valeur y.

Étant donnée une fonction de coût L, nous cherchons donc f qui minimise ce coût sur l'ensemble des valeurs possibles de $\vec{x} \in \mathbb{R}^p$, ce qui est formalisé par la notion de **risque.** Nous supposons que les couples $(\vec{x}^{\,i},y^i)$ sont les réalisations de n vecteurs aléatoires de même loi qu'un couple de variables aléatoire (X,Y).

Dans le cadre d'un problème d'apprentissage supervisé, on appelle **risque** d'un modèle h l'espérance de son coût :

$$\mathcal{R}(h) = \mathbb{E}(L(h(X),Y)). \tag{7.3}$$

Nous cherchons donc un modèle f tel que

$$f \in \arg\min_{h \in \mathcal{F}} \mathbb{E}(L(h(X),Y)). \tag{7.4}$$

Ce problème est généralement insoluble sans plus d'hypothèses : nous ne connaissons que n réalisations du couple (X,Y). On approchera donc le risque par son estimation sur ces réalisations.

On appelle **risque empirique** de h l'estimée du risque de h défini par

$$R_n(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(\vec{x}^{\,i}),y^i). \tag{7.5}$$

On appelle donc modèle obtenu par **minimisation du risque empirique** une fonction

$$f \in \arg\min_{h \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(h(\vec{x}^{\,i}),y^i). \tag{7.6}$$

Selon le choix de $\mathcal{F}$ et L, l'équation 7.6 peut avoir une solution analytique explicite. Cela ne sera pas souvent le cas ; cependant on choisira souvent une fonction de coût convexe afin de résoudre plus facilement ce problème d'optimisation.

La minimisation du risque empirique est généralement un problème *mal posé* au sens de Hadamard, c'est-à-dire qu'il n'admet pas une solution unique dépendant de façon continue des conditions initiales. Il se peut par exemple qu'un nombre infini de solutions minimise le risque empirique à zéro (voir figure 7.2).

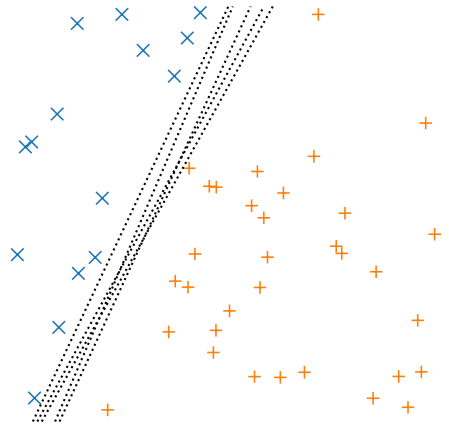


FIGURE 7.2 – Une infinité de droites séparent parfaitement les points positifs (+) des points négatifs (x). Chacune d'entre elles a un risque empirique nul.

Convergence La loi des grands nombres nous garantit que le risque empiruqe d'un modèle $h \in \mathcal{F}$ converge vers le risque quand la taille de l'échantillon tend vers l'infini :

$$R_n(h) \xrightarrow[n\to\infty]{} \mathcal{R}(h). \tag{7.7}$$

Cela ne suffit cependant pas à garantir que le minimum du risque empirique $\min_{h\in\mathcal{F}} R_n(h)$ converge vers le minimum du risque $\min_{h\in\mathcal{F}} \mathcal{R}(h)$. En effet, si $\mathcal{F}$ est l'espace des fonctions mesurables, le minimiseur de $R_n(h)$ vaut généralement 0, ce qui n'est pas le cas de $\mathcal{R}(h)$. **Il n'y a donc aucune garantie qu'un modèle qui minimise le risque empirique minimise le risque.** C'est une remarque très importante car elle signifie que le fait qu'un modèle minimise l'erreur sur nos n observations ne donne aucune garantie quant à sa performance sur d'autres individus. Nous reviendrons sur ce sujet lors du prochain chapitre, en abordant les notions de généralisation et de surapprentissage.

La convergence de la minimisation du risque empirique dépend de $\mathcal{F}$. L'étude de cette convergence est l'un des principaux éléments de la théorie de l'apprentissage de Vapnik-Chervonenkis, qui dépasse largement le cadre de ce cours.

7.4 Fonctions de coût

Il existe de nombreuses fonctions de coût. Le choix d'une fonction de coût dépend d'une part du problème en lui-même, autrement dit de ce que l'on trouve pertinent pour le cas pratique considéré, et d'autre part de considérations pratiques : peut-on ensuite résoudre le problème d'optimisation qui résulte de ce choix de façon suffisamment exacte et rapide ?Cette section présente quelques-unes des fonctions de coût les plus utilisées.

7.4.1 Coût 0/1 pour la classification binaire

Dans le cas d'une fonction f à valeurs binaires, on appelle **fonction de coût 0/1**, ou *0/1 loss*, la fonction suivante :

$$L_{0/1} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$
$$y, f(\vec{x}) \mapsto \begin{cases} 1 & \text{si } f(\vec{x}) \neq y \\ 0 & \text{sinon.} \end{cases}$$

Le risque empirique d'un modèle h sur un jeu de données est alors le nombre d'erreurs de prédiction

sur ce jeu de données.

7.4.2 Coût logistique et entropie croisée

Considérons maintenant que f est une fonction de décision à valeurs réelles. On appelle **fonction de coût logistique**, ou *logistic loss*, la fonction suivante :

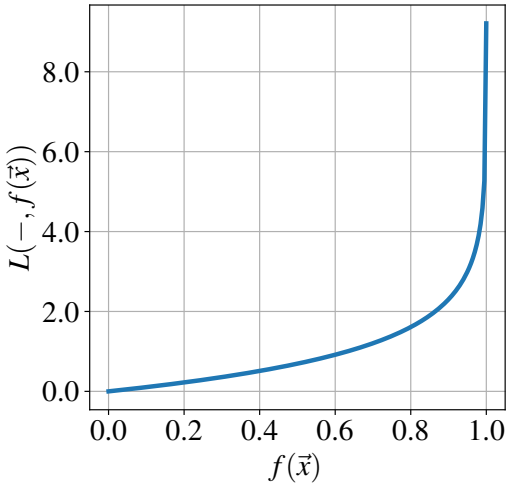
$$\begin{aligned} L_{\log} : \{-1,1\} \times \mathbb{R} &\to \mathbb{R} \\ y, f(\vec{x}) &\mapsto \log\left(1 + \exp(-y f(\vec{x}))\right). \end{aligned} \tag{7.8}$$

Si f est à valeurs dans $]0,1[$, en particulier si $f(\vec{x})$ est la probabilité que $\vec{x}$ appartienne à la classe positive, cette fonction de coût est équivalente à l'**entropie croisée**, définie pour $\mathcal{Y} = \{0,1\}$.

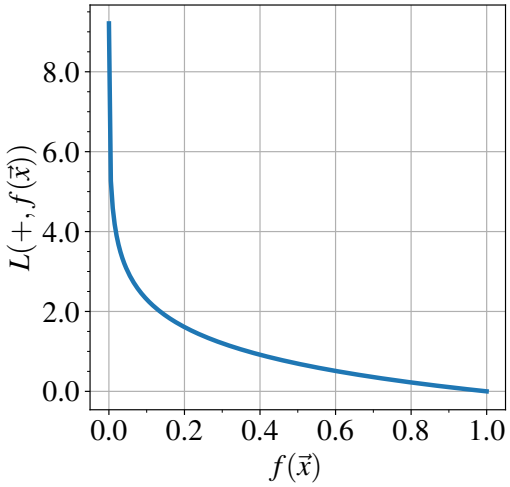
Dans le cas binaire, on appelle **entropie croisée**, ou *cross-entropy*, la fonction suivante :

$$\begin{aligned} L_H : \{0,1\} \times]0,1[&\to \mathbb{R} \\ y, f(\vec{x}) &\mapsto -y \log f(\vec{x}) - (1-y)\log(1 - f(\vec{x})). \end{aligned} \tag{7.9}$$

La figure 7.4 illustre la valeur de la fonction de coût logistique en fonction de l'étiquette y de l'individu $\vec{x}$ et de la valeur de la fonction de décision $f(\vec{x})$.



(A) Fonction de coût logistique pour un individu d'étiquette négative, en fonction de la valeur de la fonction de décision. Cette perte est d'autant plus grande que la fonction de décision est proche de 1.



(B) Fonction de coût logistique pour un individu d'étiquette positive, en fonction de la valeur de la fonction de décision. Cette perte est d'autant plus grande que la fonction de décision est proche de 0.

FIGURE 7.3 – Perte logistique / entropie croisée.

FIGURE 7.4 – Valeur de l'entropie croisée en fonction de la valeur de la fonction de décision.

★ **Pourquoi « entropie croisée » ?** L'entropie croisée est issue de la théorie de l'information, d'où son nom. En considérant que la véritable classe de $\vec{x}$ est modélisée par une distribution Q, et sa classe prédite par une distribution P, nous allons chercher à modéliser P de sorte qu'elle soit la plus proche

possible de Q. On utilise pour cela la divergence de Kullback-Leibler, définie par :

$$\begin{aligned}\mathrm{KL}(Q||P) &= \sum_{c=0,1} Q(y=c|\vec{x}) \log \frac{Q(y=c|\vec{x})}{P(y=c|\vec{x})} \\ &= -\sum_{c=0,1} Q(y=c|\vec{x}) \log P(y=c|\vec{x}) + \sum_{c=0,1} Q(y=c|\vec{x}) \log Q(y=c|\vec{x}).\end{aligned}$$

Comme $Q(y = c|\vec{x})$ vaut soit 0 (c n'est pas la classe de $\vec{x}$) soit 1 (dans le cas contraire), le deuxième terme de cette expression est nul et on retrouve ainsi la définition ci-dessus de l'entropie croisée.

7.4.3 Coût quadratique pour la régression

On appelle **fonction de coût quadratique**, ou *quadratic loss*, ou encore *squared error*, la fonction suivante :

$$\begin{aligned} L_{\mathrm{SE}} : \mathbb{R} \times \mathbb{R} &\to \mathbb{R} \\ y, f(\vec{x}) &\mapsto \frac{1}{2}\left(y - f(\vec{x})\right)^2 . \end{aligned} \tag{7.10}$$

Le coefficient $\frac{1}{2}$ permet d'éviter d'avoir des coefficients multiplicateurs quand on dérive le risque empirique pour le minimiser.

7.5 Apprentissage supervisé d'un modèle paramétrique

7.5.1 Modèles paramétriques

On parle de **modèle paramétrique** quand l'espace des hypothèses $\mathcal{F}$ est un ensemble de fonctions définies par une expression analytique paramétrisée par un nombre fini de paramètres.

C'est le cas de l'espace des hypothèses défini plus haut par l'équation (7.2) : les paramètres sont au nombre de 4 et il s'agit de α, β, a, et b. Le but de l'apprentissage sera de déterminer les valeurs de ces paramètres.

À l'inverse, la méthode du plus proche voisin, qui associe à $\vec{x}$ l'étiquette du point du jeu d'entraînement dont il est le plus proche en distance euclidienne, apprend un modèle non paramétrique : on ne sait pas écrire la fonction de décision comme une fonction des variables prédictives.

Nous verrons au chapitre 9 des exemples de modèles non paramétriques, et nous concentrons maintenant sur les modèles de régression paramétriques.

Nous considérons pour la suite de ce chapitre disposer d'un jeu $\mathcal{D} = \{\vec{x}^{\,i}, y^i\}_{i=1,\dots,n}$ de n observations en p dimensions et leurs étiquettes réelles. Nous considérons comme espace des hypothèses un ensemble de modèles paramétrisés par un vecteur $\vec{\beta} \in \mathbb{R}^m$.

7.5.2 Minimisation du risque empirique

Si nous utilisons comme fonction de coût le coût quadratique défini par l'équation (7.10), la minimisation du risque empirique comme définie par l'équation (7.6) consiste à trouver

$$\vec{\beta}^{\,*} \in \underset{\vec{\beta}\in\mathbb{R}^m}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} (f_{\vec{\beta}}(\vec{x}^{\,i}) - y^i)^2. \tag{7.11}$$

C'est ce que l'on appelle la **minimisation des moindres carrés**, une méthode bien connue depuis Gauss et Legendre.

7.5.3 Formulation probabiliste des régressions paramétriques ★

Nous supposons comme précédemment que les couples $(\vec{x}^{\,i},y^i)$ sont les réalisations de n vecteurs aléatoires de même loi qu'un couple de variables aléatoire (X,Y).

Cela revient à supposer que la relation entre X et Y peut s'écrire comme

$$Y = f_{\vec{\beta}}(X) + \epsilon. \tag{7.12}$$

Faisons maintenant **l'hypothèse d'un bruit gaussien centré en** 0 **:** le terme d'erreur ϵ est normalement distribué, centré en 0.

$$\epsilon \sim \mathcal{N}(0,\, \sigma^2). \tag{7.13}$$

L'équation (7.12) revient alors à

$$Y|X = \vec{x} \sim \mathcal{N}\left(f_{\vec{\beta}}(\vec{x}),\, \sigma^2\right). \tag{7.14}$$

Exemple

L'équation (7.14) est illustrée sur la figure 7.5 dans le cas où $p = 1$ et l'espace des hypothèses est l'ensemble des fonctions linéaires d'une variable : $\mathcal{F} = \{x \mapsto f_{\alpha,\beta}(x) = \alpha x + \beta \ ; (\alpha,\beta) \in \mathbb{R}^2\}$. La distribution des valeurs de l'étiquette d'un individu x^* selon le modèle $f_{\alpha,\beta}$ est une gaussienne centrée en $f_{\alpha,\beta}(x^*)$. Sa densité est notée $g_{Y|X=x^*}$.

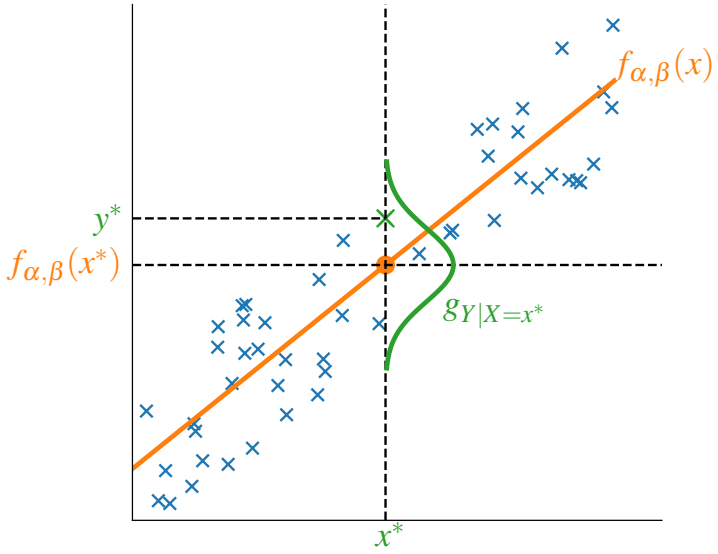


FIGURE 7.5 – Pour une observation x^* donnée (ici en une dimension) , la distribution des valeurs possibles de l'étiquette correspondante est une gaussienne centrée en $f(x^*)$. La vraie valeur de l'étiquette est y^*.

7.5.4 Estimation par maximum de vraisemblance ★

Sous l'hypothèse (7.14), nous pouvons donc estimer $\vec{\beta}$ en maximisant la log-vraisemblance de l'échantillon $\left((\vec{x}^{\,1},y^1),(\vec{x}^{\,2},y^2),\ldots,(\vec{x}^{\,n},y^n)\right)$, considéré comme la réalisation de l'échantillon aléatoire $((X_1,Y_1),(X_2,Y_2),\ldots,(X_n,Y_n))$, lui même constitué de n copies i.i.d. de (X,Y).

En notant $g_{X,Y}$ la densité jointe de (X,Y) ; $g_{Y|X=x}$ la densité de $Y|X=x$; et g_X la densité de X, cette log-vraisemblance s'écrit

$$\ell\left((\vec{x}^{\,1},y^1),(\vec{x}^{\,2},y^2),\ldots,(\vec{x}^{\,n},y^n);\vec{\beta}\right) = \log\prod_{i=1}^{n} g_{X,Y}(\vec{x}^{\,i},y^i) = \log\prod_{i=1}^{n} g_{Y|X=\vec{x}^{\,i}}(y^i) + \log\prod_{i=1}^{n} g_X(\vec{x}^{\,i})$$
$$= -\log\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y^i - f_{\vec{\beta}}(\vec{x}^{\,i})\right)^2 + \mathcal{C}.$$

Dans cette dernière équation, $\mathcal{C}$ est une constante par rapport à $\vec{\beta}$, et provient d'une part du coefficient $\frac{1}{\sqrt{2\pi}}$ de la distribution normale et d'autre part des $g_X(\vec{x}^{\,i})$.

Ainsi, maximiser la vraisemblance dans ce contexte de bruit gaussien centré revient à minimiser

$$\sum_{i=1}^{n}\left(y^i - f_{\vec{\beta}}(\vec{x}^{\,i})\right)^2.$$

On retrouve ici la méthode des moindres carrés de l'équation (7.11).

7.6 Régression linéaire

Nous allons maintenant appliquer la minimisation des moindres carrés au cas où $\mathcal{F}$ est l'ensemble des fonctions linéaires de p variables.

7.6.1 Formulation

Nous choisissons une fonction de décision f de la forme

$$f : \vec{x} \mapsto \beta_0 + \sum_{j=1}^{p}\beta_j x_j. \tag{7.15}$$

Ici, $\vec{\beta} \in \mathbb{R}^{p+1}$ et donc $m = p+1$.

7.6.2 Solution

On appelle **régression linéaire** le modèle de la forme $f : \vec{x} \mapsto \beta_0 + \sum_{j=1}^{p}\beta_j x_j$ dont les coefficients sont obtenus par minimisation de la somme des moindres carrés, à savoir :

$$\arg\min_{\vec{\beta}\in\mathbb{R}^{p+1}} \sum_{i=1}^{n}\left(y^i - \left(\beta_0 + \sum_{j=1}^{p}\beta_j x_j\right)\right)^2. \tag{7.16}$$

Nous pouvons réécrire le problème 7.16 sous forme matricielle, en ajoutant à gauche à la matrice d'observations $X \in \mathbb{R}^p$ une colonne de 1 :

$$X \leftarrow \begin{pmatrix} 1 & x_1^1 & \cdots & x_p^1 \\ \vdots & \vdots & \ldots & \vdots \\ 1 & x_1^n & \cdots & x_p^n \end{pmatrix}. \tag{7.17}$$

La somme des moindres carrés s'écrit alors

$$\text{RSS} = \left(\vec{y} - X\vec{\beta}\right)^\top \left(\vec{y} - X\vec{\beta}\right). \tag{7.18}$$

Il s'agit d'une forme quadratique convexe en $\vec{\beta}$, que l'on peut donc minimiser en annulant son gra-

dient $\nabla_{\vec{\beta}}\text{RSS} = -2X^\top \left(\vec{y} - X\vec{\beta}\right)$. La somme des moindres carrés est minimale si $\vec{\beta}$ vérifie

$$X^\top X\vec{\beta} = X^\top \vec{y}. \tag{7.19}$$

Solution explicite Si le rang de la matrice X est égal à son nombre de colonnes, alors $X^\top X$ est inversible et la somme des moindres carrés de l'équation (7.18) est minimisée pour

$$\vec{\beta}^* = \left(X^\top X\right)^{-1} X^\top \vec{y}.$$

Si $X^\top X$ n'est pas inversible, on pourra néanmoins trouver une solution (non unique) pour $\vec{\beta}$ en utilisant à la place de $\left(X^\top X\right)^{-1}$ un pseudo-inverse (par exemple, celui de Moore-Penrose) de $X^\top X$, c'est-à-dire une matrice M telle que $X^\top XMX^\top X = X^\top X$.

Méthode de descente On peut aussi (et ce sera préférable quand p est grand et que l'inversion de la matrice $X^\top X \in \mathbb{R}^{p\times p}$ est donc coûteuse) obtenir une estimation de $\vec{\beta}$ par un algorithme à directions de descente.

Interprétation La régression linéaire produit un modèle interprétable, au sens où les β_j permettent de comprendre l'importance relative des variables sur la prédiction. En effet, plus $|\beta_j|$ est grande, plus la j-ème variable a un effet important sur la prédiction, et le signe de β_j nous indique la direction de cet effet.

Attention ! Cette interprétation n'est valide que si les variables ne sont pas corrélées, et que x_j peut être modifiée sans perturber les autres variables. De plus, si les variables sont corrélées, X n'est pas de rang colonne plein et $X^\top X$ n'est donc pas inversible. Ainsi la régression linéaire admet plusieurs solutions. Intuitivement, on peut passer de l'une à l'autre de ces solutions car une perturbation d'un des poids β_j peut être compensée en modifiant les poids des variables corrélées à x_j.

Remarque Nous avons traité ici de problèmes de *régression* uniquement. Nous traiterons de classification paramétrique dans la PC 6.