



Data mining

II. Modélisation Statistique & Apprentissage

PHILIPPE BESSE

Version janvier 2003 — mises à jour : www.lsp.ups-tlse.fr/Besse

Laboratoire de Statistique et Probabilités — UMR CNRS C5583
Université Paul Sabatier — 31062 – Toulouse cedex 4.

Avant-propos

Motivations du *data mining*

Le développement des moyens informatiques de stockage (bases de données) et de calcul permet le traitement et l'analyse d'ensembles de données très volumineux. Plus récemment, le perfectionnement des interfaces offrent aux utilisateurs, statisticiens ou non, des possibilités de mise en œuvre très simples des outils logiciels. Cette évolution, ainsi que la popularisation de nouvelles méthodes algorithmiques (réseaux de neurones) et outils graphiques, conduit au développement et à la commercialisation de logiciels intégrant un sous-ensemble de méthodes statistiques et algorithmiques sous la terminologie de *Data Mining* : la prospection ou *fouille de données*. Cette approche, issue du marketing spécialisé dans la gestion de la relation client (*client relation management* ou CRM) trouve également des développements et applications industrielles en contrôle de qualité ou même dans certaines disciplines scientifiques dès lors que les ingénieurs et chercheurs sont confrontés à un volume de données important. Besse et col. (2001) présente une introduction détaillée de cette démarche et des relations qu'elle entretient avec les disciplines traditionnelles Statistique et Informatique. L'accroche publicitaire souvent citée par les éditeurs de logiciels (SAS) est :

Comment trouver un diamant dans un tas de charbon sans se salir les mains.

Nous proposons d'évaluer et d'expérimenter la réalité de cette annonce qui s'adresse à un marché en pleine expansion. Les entreprises sont en effet très motivées pour tirer parti et amortir, par une aide à la décision quantifiée, les coûts de stockage des teras octets que leur service informatique s'emploie à administrer.

Le contexte informationnel de la fouille de données est celui des *data wharehouses*. Un entrepôt de données, dont la mise en place est assuré par un gestionnaire de données (data manager) est un ensemble de bases relationnelles extraites des données brutes de l'entreprise et relatives à une problématique :

- gestion des stocks (flux tendu), des ventes d'un groupe afin de prévoir et anticiper au mieux les tendances du marché,
- suivi des fichiers clients d'une banque, d'une assurance, associés à des données socio-économiques (INSEE), à l'annuaire, en vue de la constitution d'une segmentation (typologie) pour cibler des opérations de marketing ou des attributions de crédit. La *gestion de la relation client* vise à une individualisation ou personnalisation de la production et de la communication afin d'évacuer la notion de *client moyen*.
- recherche, spécification puis ciblage de *niches* de marché les plus profitables (banque) ou au contraire les plus risquées (assurance) ;
- suivi en ligne des paramètres de production en contrôle de qualité pour détecter au

- plus vite l'origine d'une défaillance ;
- prospection textuelle (*text mining*) et veille technologique ;
- *web mining* et comportement des internautes ;
- ...

Cet environnement se caractérise par

- une informatique hétérogène faisant intervenir des sites distants (Unix, Dos, NT, VM. . .) à travers le réseau de l'entreprise (intranet) ou même des accès extérieurs (internet). Des contraintes d'efficacité, de fiabilité ou de sécurité conduisent à répartir, stocker l'information à la source plutôt qu'à la dupliquer systématiquement ou à la centraliser.
- L'incompatibilité logique des informations observées sur des échantillons différents ne présentant pas les mêmes strates, les mêmes codifications.
- Des volumes et flux considérables de données issues de saisies automatisées et chiffrés en téra-octets.
- La nécessité de ne pas exclure *a priori* un traitement *exhaustif* des données afin de ne pas laisser échapper, à travers le crible d'un *sondage*, des groupes de faibles effectifs mais à fort impact économique.

Stratégie du *data mining*

Dans tout ce qui suit, nous disposons d'un ensemble d'observations. Les caractéristiques ou variables $X = (X^1, \dots, X^p)$ dites explicatives ont été observées sur un ensemble de n objets, individus ou unités statistiques. Un premier travail, souvent fastidieux mais incontournable, consiste à mener une exploration statistique de ces données : allure des distributions, présence de données atypiques, corrélations et cohérence, transformations éventuelles des données, description multidimensionnelle, classification. C'est l'objet de la première partie de ce document. La deuxième partie décrit les outils de modélisation statistique ou encore d'apprentissage utilisables pour la prédiction d'une variable *cible* Y par les variables explicatives X^j .

L'enchaînement de ces étapes (exploration puis apprentissage) constitue le fondement de la fouille de données.

Pour comprendre la structure et bien appréhender le contenu de ce cours, il est important d'intégrer rapidement ce qu'est la stratégie à mettre en œuvre pour aboutir au bon *apprentissage* ou encore au bon *modèle prédictif* recherché à partir des données observées.

Attention, il faut bien noter que, contrairement à une démarche statistique traditionnelle dans laquelle l'observation des données est intégrée à la méthodologie (plannification de l'expérience), les données sont ici *préalables* à l'analyse. Néanmoins il est clair que les préoccupations liées à leur analyse et à son objectif doivent intervenir le plus en amont possible pour s'assurer quelques chances de succès.

Les étapes de la fouille de données :

- i. Extraction des données avec ou sans échantillonnage faisant référence à des techniques de sondage appliquées ou applicables à des bases de données.
- ii. Exploration des données pour la détection de valeurs aberrantes ou seulement atypiques, d'incohérences, pour l'étude des distributions des structures de corrélation, recherche de typologies, pour des transformations des données. . .

-
- iii. Partition aléatoire de l'échantillon (apprentissage, validation, test) en fonction de sa taille et des techniques qui seront utilisées pour estimer une erreur de prédiction en vue des choix de modèle, choix et certification de méthode.
 - iv. Pour chacune des méthodes considérées : modèle linéaire général (gaussien, binomial ou poissonien), discrimination paramétrique (linéaire ou quadratique) ou non paramétrique, k plus proches voisins, arbre, réseau de neurones (perceptron), support vecteur machine, combinaison de modèles (bagging, boosting).
 - estimer le modèle pour une valeur donnée d'un paramètre de *complexité* : nombre de variables, de voisins, de feuilles, de neurones, , durée de l'apprentissage, largeur de fenêtre. . . ;
 - optimiser ce paramètre (sauf pour les combinaisons de modèles affranchies des problèmes de sur-apprentissage) en fonction de la technique d'estimation de l'erreur retenue : échantillon de validation, validation croisée, approximation par pénalisation de l'erreur d'ajustement.
 - v. Comparaison des modèles optimaux obtenus (un par méthode) par estimation de l'erreur de prévision sur l'échantillon test ou, si la présence d'un échantillon test est impossible, sur le critère de pénalisation de l'erreur (Akaike par exemple) s'il en existe une version pour chacune des méthodes considérées.
 - vi. Itération éventuelle de la démarche précédente (validation croisée), si l'échantillon test est trop réduit, depuis (iii). Partitions aléatoires successives de l'échantillon pour moyenner sur plusieurs cas l'estimation finale de l'erreur de prédiction et s'assurer de la robustesse du modèle obtenu.
 - vii. Choix de la méthode retenue en fonction de ses capacités de prédiction, de sa robustesse mais aussi, éventuellement, de l'interprétabilité du modèle obtenu.

Objectif

L'objet de ce cours est d'introduire, sous une forme homogène et synthétique, les principales techniques d'exploration, de modélisation ou encore d'apprentissage utilisées le plus couramment en fouille de données et citées dans la section précédente. Il a fallu faire des choix dans l'ensemble des techniques proposées et leurs nombreux avatars. La forme et le contenu sont guidés par les besoins exprimés lors des stages réalisées par les étudiants du DESS de Statistique et Econométrie¹ ou encore par les thèmes des collaborations industrielles du laboratoire de Statistique et Probabilités². Remarquons que les principaux logiciels commerciaux (SAS, Splus, SPSS) ou gratuits (R), performants et s'imposant par des interfaces très conviviales (Enterprise Miner, Insightfull Miner, Clementine), contribuent largement à la diffusion, voire la pénétration, de méthodes très sophistiquées dans des milieux imperméables à une conceptualisation mathématique trop abstraite.

Le choix a été fait de conserver et expliciter, dans la mesure du possible, les concepts originaux de chaque méthode dans son cadre disciplinaire tout en tâchant d'homogénéiser notations et terminologies. L'objectif principal est de faciliter la compréhension et l'interprétation des techniques des principaux logiciels pour en faciliter une *utilisation pertinente et réfléchie*. Un exemple élémentaire de recherche d'un score d'appétance issu du

¹<http://www.univ-tlse1.fr/formation/DESS/DESS-StatEconometrie.html>

²<http://www.lsp.ups-tlse.fr>

marketing bancaire illustre les différents points abordés. Traité avec les logiciels SAS, Splus ou R, il sert de “fil rouge” tout au long du cours.

Chapitre 1

Introduction

1 Objectif

Dès qu'un phénomène, qu'il soit physique, biologique ou autre, est trop complexe ou encore trop bruyé pour accéder à une description analytique débouchant sur une modélisation déterministe, un ensemble d'approches ont été élaborées afin d'en décrire au mieux le comportement à partir d'une série d'observations. Citons la reconnaissance de la parole ou de caractères manuscrits, l'imagerie médicale ou satellitaire, la prévision d'une grandeur climatique ou économique, du comportement d'un client... la plupart des disciplines scientifiques sont concernées. Historiquement, la Statistique s'est beaucoup développée autour de ce type de problèmes et a proposé des *modèles* incorporant d'une part des *variables explicatives* et, d'autre part, une composante aléatoire ou *bruit*. Il s'agit alors d'*estimer* les *paramètres* du modèle à partir des observations. Dans la même situation, la communauté informatique parle plutôt d'*apprentissage* visant le même objectif. Apprentissage machine (machine learning) ou reconnaissance de forme (pattern recognition) en sont les principaux mots-clefs.

2 Problématique

2.1 Supervisé *vs.* non-supervisé

Distinguons ensuite deux types de problèmes : la présence ou non d'une variable à *expliquer* Y ou d'une *forme* à reconnaître qui a été, conjointement avec X , observée sur les mêmes objets. Dans le premier cas il s'agit bien d'un problème de modélisation ou apprentissage supervisé : trouver une fonction ϕ susceptible, au mieux selon un critère à définir, de reproduire Y ayant observé X .

$$Y = \phi(X) + \varepsilon$$

où ε symbolise le bruit ou erreur de mesure avec le parti pris le plus commun que cette erreur est additive.

Dans le cas contraire, en l'absence d'une variable à expliquer, il s'agit alors d'apprentissage dit non-supervisé. L'objectif généralement poursuivi est la recherche d'une typologie ou taxinomie des observations : comment regrouper celles-ci en classes homogènes mais les plus dissemblables entre elles. C'est un problème de classification (*clustering*).

Attention, l'anglais *classification* se traduit plutôt en français par discrimination ou classement (apprentissage supervisé) tandis que la recherche de classes (*clustering*) (apprentissage non-supervisé) fait appel à des méthodes de classification ascendante hiérarchique ou à des algorithmes de réallocation dynamique (*k*-means) ou de carte auto-organisatrices (Kohonen). Ces méthodes de classification ou *clustering* ne sont pas abordées ici, elles ont été regroupées avec les techniques exploratoires de la première partie (Baccini et Besse 2000).

2.2 Modélisation *vs.* apprentissage

Tout au long de ce document, les termes de *modélisation* et d'*apprentissage* sont utilisés comme des synonymes ce qui est abusif tant que les objectifs d'une étude n'ont pas été clairement explicités. Dans la tradition statistique, la notion de *modèle* est centrale surtout avec une finalité *explicative*. Il s'agit alors d'approcher la réalité, le *vrai* modèle, éventuellement basé sur une théorie physique, économique... sous-jacente. Le choix du modèle (cf. ci-dessous) est alors guidé par des critères d'*ajustement* et les décisions de validité, de présence d'effets, basées sur des *tests* reposant sur des hypothèses probabilistes. L'interprétation du rôle de chaque variable explicative est prépondérante dans la démarche.

En revanche, dans un but *prédictif* apparaît que le meilleur modèle n'est pas nécessairement le vrai. La théorie de l'*apprentissage* (Vapnik, 1999) montre alors que le cadre théorique est différent et les majorations d'erreur requièrent une autre approche. Les choix sont basés sur des critères de qualité de prédiction visant à la recherche de *modèles parcimonieux* dont l'interprétabilité passe au deuxième plan.

2.3 Discrimination *vs.* régression

Le type des variables statistiques considérées diffèrent selon l'espace dans lequel elles prennent leurs valeurs. Elles peuvent être quantitatives à valeurs réelles¹ ou qualitatives à valeurs dans un ensemble de cardinal fini. Certaines méthodes d'apprentissage ou de modélisation s'adaptent à tout type de variables explicatives tandis que d'autres sont spécialisées. Enfin, si *Y* à expliquer est qualitative, on parle de discrimination, classement ou reconnaissance de forme tandis que si *Y* est quantitative on parle, par habitude, d'un problème de régression. Dans ce cas encore, certaines méthodes sont spécifiques (régression linéaire, analyse discriminante) tandis que d'autres s'adaptent sans modification profonde remettant en cause leur principe (réseaux de neurones, arbres...).

2.4 Statistique, informatique et taille des données

Lorsque des hypothèses relatives au modèle (linéarité) et aux distributions sont vérifiées c'est-à-dire, le plus souvent, lorsque l'échantillon ou les résidus sont supposés suivre des lois se mettant sous la forme d'une famille exponentielle (gaussienne, binomiale, poisson...), les techniques statistiques de modélisation tirées du modèle linéaire général sont optimales et, surtout dans le cas d'échantillons de taille restreinte, il semble difficile de faire mieux.

¹Le traitement de données fonctionnelles (Besse et Cardot, 2003), c'est-à-dire l'étude de courbes, nécessite généralement une décomposition préalable sur une base appropriée (vecteurs propres, fourier, ondelettes) avec, selon le cas, lissage ou interpolation avant de pouvoir mettre en œuvre les techniques spécifiques d'apprentissage. Ces aspects ne sont pas abordés.

En revanche, dès que les hypothèses distributionnelles ne sont pas vérifiées, dès que les relations supposées entre les variables ne sont pas linéaires ou encore dès que le volume des données est important, d'autres méthodes viennent concurrencer l'approche statistique classique.

Prenons un exemple simple : expliquer une variable quantitative Y par un ensemble $\{X^1, \dots, X^p\}$ de variables également quantitatives :

$$Y = \phi(X^1, \dots, X^p) + \varepsilon.$$

observées sur un échantillon $(y_i, \mathbf{x}_i); i = 1, \dots, n$ de taille n . Si ϕ est supposée linéaire et p petit, de l'ordre d'une dizaine ; le problème est bien connu et largement débattu dans la littérature. Dans le cas où ϕ n'est pas franchement linéaire et n grand, il est possible d'estimer précisément un nombre plus important de paramètres et donc d'envisager des modèles plus sophistiqués. Si on s'en tient au modèle gaussien usuel, même le cas le plus simple d'un modèle polynômial devient vite problématique. En effet, lorsque ϕ est linéaire, prenons $p = 10$, la procédure de choix de modèle est confrontée à un ensemble de 2^{10} modèles possibles et des algorithmes astucieux permettent encore de s'en sortir. En revanche, considérer pour ϕ un simple polynôme du deuxième voire troisième degré avec toutes ses interactions, amène à considérer un nombre considérable de paramètres et donc, par explosion combinatoire, un nombre astronomique de modèles possibles. D'autres méthodes doivent alors être considérées en prenant en compte nécessairement la complexité algorithmique des calculs. Ceci explique l'implication d'une autre discipline, l'informatique, dans cette problématique. Le souci de calculabilité l'emporte sur la définition mathématique du problème qui se ramène à l'optimisation d'un critère d'ajustement de ϕ sur un ensemble de solutions plus ou moins riche. Ces méthodes ont souvent été développées dans un autre environnement disciplinaire : informatique, intelligence artificielle. . . ; k plus proches voisins, réseaux de neurones, arbres de décisions, *support vector machine* deviennent des alternatives crédibles dès lors que le nombre d'observations est suffisant.

2.5 Choix de méthode

Avec l'avènement du *data mining*, de très nombreux articles comparent et opposent les techniques sur des jeux de données publics et proposent des améliorations incrémentales de certains algorithmes. Après une période fiévreuse où chacun tentait d'afficher la suprématie de sa méthode, un consensus s'est établi autour de l'idée qu'il n'y a pas de "meilleure méthode". Chacune est plus ou moins bien adaptée au problème posé, à la nature des données ou encore aux propriétés de la fonction ϕ à approcher ou estimer. Sur le plan méthodologique, il est alors important de savoir comparer des méthodes afin de choisir la plus pertinente. Cette comparaison repose sur une estimation d'erreur (de régression ou de classement) qu'il est nécessaire de conduire avec soin. Un chapitre (3) est consacré à ce point.

2.6 Choix de modèle : équilibre biais-variance

Tous les auteurs s'accordent pour souligner l'importance qu'il y a à construire des modèles parcimonieux quelque soit la méthode utilisée. Toutes les méthodes sont concernées : nombre de variables explicatives, de feuilles dans un arbre ou de neurones dans une couche cachée. . . Seuls les algorithmes de combinaison de modèles (bagging, boosting)

contournent cette étape au prix d'un accroissement sensible du volume des calculs et de l'interprétabilité des résultats obtenus.

L'alternative est claire, plus un modèle est complexe et donc plus il intègre de paramètres et plus il est capable de s'ajuster aux données et donc d'engendrer une erreur faible d'ajustement. En revanche, un tel modèle peut s'avérer défaillant lorsqu'il s'agira de prévoir ou généraliser, c'est-à-dire de s'appliquer à des données qui n'ont pas participé à son estimation. Exemple : discriminer dans \mathbb{R}^2 une frontière quadratique par une régression linéaire ou par un polynôme de degré plus élevé.

Ce problème s'illustre aussi facilement en régression classique. Ajouter des variables explicatives dans un modèle ne peut que réduire l'erreur d'ajustement (le R^2) et réduit le biais si le "vrai" modèle est un modèle plus complet. Mais, ajouter des variables fait également croître la variance des estimateurs et donc celle des prédictions qui se dégradent rapidement avec la multicolinéarité des variables explicatives. Un risque pour le modèle, ou erreur quadratique de prédiction, s'exprimant comme le carré du biais plus la variance, il est important d'optimiser le dosage entre biais et variance en contrôlant le nombre de variables dans le modèle afin de minimiser le risque. Ces remarques conduisent à la définition de critères de choix de modèle dont le C_p de Mallows fut un précurseur en régression suivi par d'autres propositions : Akaike (AIC), Schwartz (BIC)...

Plus que celui de la méthode, le choix du bon modèle ou de la bonne complexité de celui-ci dans une classe de méthodes données est primordial. En conséquence, les problèmes d'optimisation considérés doivent mettre en œuvre un critère qui prend en compte la *complexité du modèle*, c'est-à-dire la complexité de l'espace dans lequel la solution est recherchée.

2.7 Choix de modèle : sélection *vs.* régularisation

Selon la méthode considérée, la complexité du modèle s'exprime de différentes façons. Simple par sélection de variable en régression linéaire, la complexité est directement liée à la dimension de l'espace engendré et donc au nombre de variables. Les choses se compliquent pour les modèles non-linéaires lorsque, à dimension fixée, c'est la plus ou moins grande flexibilité des solutions qui doit être pénalisée.

C'est typiquement le cas en régression non-paramétrique ou fonctionnelle. Une pénalisation faisant intervenir la norme carrée de la dérivée seconde contrôle la flexibilité d'un lissage spline. La "largeur de fenêtre" du noyau contrôle également la régularité de la solution. En régression linéaire, si le nombre et les variables sont déterminés, la version "ridge" de la régression pénalise la norme carrée du vecteur des paramètres et restreint ainsi, par régularisation, l'espace des solutions pour limiter l'effet de la multicolinéarité.

Enfin, pour aborder en toute généralité les situations les plus compliquées, Vapnik (1999) a formalisé la théorie de l'apprentissage en introduisant une notion particulière de dimension pour toute famille de modèles.

2.8 Contenu

Chaque méthode ou famille de méthodes de modélisation et d'apprentissage parmi les plus répandues, est présentée de façon plus ou moins succincte dans un chapitre distinct avec un objectif prédictif. La régression linéaire classique en statistique prend une

place particulière à titre pédagogique. Très antérieure aux autres, elle donne lieu à une bibliographie abondante. Conceptuellement plus simple, elle permet d'introduire plus facilement les problématiques rencontrées comme celle du choix d'un modèle par ses deux approches types : la sélection de variable ou la régularisation (*ridge*). Pour une meilleure compréhension des logiciels qui y font largement référence, une introduction (annexe) au *modèle linéaire général* fournit le cadre théorique nécessaire à l'unification des régressions linéaire et logistique ; cette dernière reste toujours très utilisée en scoring. La présentation de l'analyse discriminante décisionnelle, paramétrique ou non paramétrique, les k plus proches voisins, permet d'introduire également des notions de théorie bayésienne de la décision. Un chapitre incontournable est consacré aux techniques d'estimation d'une erreur de prédiction sur lesquelles reposent les choix opérationnels décisifs : de modèle, de méthode mais aussi l'évaluation de la précision des résultats escomptés. Les chapitres suivants sont consacrés aux techniques algorithmiques : arbres binaires de décision (*classification and regression trees* ou CART) et à celles plus directement issues de la théorie de l'apprentissage machine (*machine learning*) : réseau de neurones et perceptron, agrégation de modèles (*boosting, random forest*). Des annexes apportent des compléments théoriques : introduction au modèle linéaire général, le bootstrap.

Chapitre 2

Régression linéaire

1 Introduction

Ce chapitre ne propose qu’une introduction au modèle gaussien, à sa définition et à son estimation en privilégiant l’objectif de prédiction. Il s’attarde donc sur le problème délicat du choix de modèle afin, principalement, d’en introduire les grands principes pour les adapter au cas de la régression logistique largement utilisée en prospection de données. Une dernière section introduit le modèle d’analyse de covariance mais de nombreux aspects : colinéarité, points influents, tests, analyse de variance, modèle multinomial ou poissonien (modèle log-linéaire) . . . sont négligés et à rechercher dans la bibliographie de même qu’une présentation globale du *modèle linéaire général* incluant toutes ces approches et seulement résumée en annexe. Les statistiques des tests élémentaires sont explicitées afin de faciliter la lectures et l’interprétation des résultats issus des logiciels.

Le but premier de ce chapitre est donc l’explication ou plutôt, la modélisation dans un but prédictif, d’une variable *quantitative* par plusieurs variables quantitatives (régression linéaire multiple) ou par un mélange de variables quantitatives et qualitatives (analyse de covariance).

2 Modèle

Le modèle de régression linéaire multiple est l’outil statistique le plus habituellement mis en œuvre pour l’étude de données multidimensionnelles. Cas particulier de modèle linéaire, il constitue la généralisation naturelle de la régression simple.

Une variable quantitative Y dite à *expliquer* (ou encore, réponse, exogène, dépendante) est mise en relation avec p variables quantitatives X^1, \dots, X^p dites *explicatives* (ou encore de contrôle, endogènes, indépendantes, régresseurs).

Les données sont supposées provenir de l’observation d’un échantillon statistique de taille n ($n > p + 1$) de $\mathbb{R}^{(p+1)}$:

$$(x_i^1, \dots, x_i^j, \dots, x_i^p, y_i) \quad i = 1, \dots, n.$$

L’écriture du *modèle linéaire* dans cette situation conduit à supposer que l’espérance de Y appartient au sous-espace de \mathbb{R}^n engendré par $\{\mathbf{1}, X^1, \dots, X^p\}$ où $\mathbf{1}$ désigne le vecteur

de \mathbb{R}^n constitué de “1”. C’est-à-dire que les $(p+1)$ variables aléatoires vérifient :

$$y_i = \beta_0 + \beta_1 x_i^1 + \beta_2 x_i^2 + \cdots + \beta_p x_i^p + \varepsilon_i \quad i = 1, 2, \dots, n$$

avec les hypothèses suivantes :

- i. Les ε_i sont des termes d’erreur, d’une variable U , non observés, indépendants et identiquement distribués; $E(\varepsilon_i) = 0, \text{Var}(\varepsilon) = \sigma^2 \mathbf{I}$.
- ii. Les termes x^j sont supposés déterministes (facteurs contrôlés) **ou bien** l’erreur U est indépendante de la distribution conjointe de X^1, \dots, X^p . On écrit dans ce dernier cas que :

$$E(Y|X^1, \dots, X^p) = \beta_0 + \beta_1 X^1 + \beta_2 X^2 + \cdots + \beta_p X^p \text{ et } \text{Var}(Y|X^1, \dots, X^p) = \sigma^2.$$

- iii. Les paramètres inconnus β_0, \dots, β_p sont supposés constants.
- iv. En option, pour l’étude spécifique des lois des estimateurs, une quatrième hypothèse considère la normalité de la variable d’erreur U ($\mathcal{N}(0, \sigma^2 \mathbf{I})$). Les ε_i sont alors i.i.d. de loi $\mathcal{N}(0, \sigma^2)$.

Les données sont rangées dans une matrice $\mathbf{X}(n \times (p+1))$ de terme général x_i^j , dont la première colonne contient le vecteur $\mathbf{1}$ ($x_0^i = 1$), et dans un vecteur \mathbf{Y} de terme général y_i . En notant les vecteurs $\boldsymbol{\varepsilon} = [\varepsilon_1 \cdots \varepsilon_p]'$ et $\boldsymbol{\beta} = [\beta_0 \beta_1 \cdots \beta_p]'$, le modèle s’écrit matriciellement :

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

3 Estimation

Conditionnellement à la connaissance des valeurs des X^j , les paramètres inconnus du modèle : le vecteur $\boldsymbol{\beta}$ et σ^2 (paramètre de nuisance), sont estimés par minimisation du critère des moindres carrés (M.C.) ou encore, en supposant (iv), par maximisation de la vraisemblance (M.V.). Les estimateurs ont alors les mêmes expressions, l’hypothèse de normalité et l’utilisation de la vraisemblance conférant à ces derniers des propriétés complémentaires.

3.1 Estimation par M.C.

L’expression à minimiser sur $\boldsymbol{\beta} \in \mathbb{R}^{p+1}$ s’écrit :

$$\begin{aligned} \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i^1 - \beta_2 x_i^2 - \cdots - \beta_p x_i^p)^2 &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 \\ &= (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ &= \mathbf{y}'\mathbf{y} - 2\boldsymbol{\beta}'\mathbf{X}'\mathbf{y} + \boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta}. \end{aligned}$$

Par dérivation matricielle de la dernière équation on obtient les “équations normales” :

$$\mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{X}\boldsymbol{\beta} = 0$$

dont la solution correspond bien à un minimum car la matrice hessienne $2\mathbf{X}'\mathbf{X}$ est définie-positive.

Nous faisons l'hypothèse supplémentaire que la matrice $\mathbf{X}'\mathbf{X}$ est inversible, c'est-à-dire que la matrice \mathbf{X} est de rang $(p + 1)$ et donc qu'il n'existe pas de colinéarité entre ses colonnes. En pratique, si cette hypothèse n'est pas vérifiée, il suffit de supprimer des colonnes de \mathbf{X} et donc des variables du modèle. Des diagnostics de colinéarité et des critères aident au choix des variables.

Alors, l'estimation des paramètres β_j est donnée par :

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

et les valeurs ajustées (ou estimées, prédites) de \mathbf{y} ont pour expression :

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{b} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{H}\mathbf{y}$$

où $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ est appelée “*hat matrix*” ; elle met un chapeau à \mathbf{y} . Géométriquement, c'est la matrice de projection orthogonale dans \mathbb{R}^n sur le sous-espace $\text{Vect}(\mathbf{X})$ engendré par les vecteurs colonnes de \mathbf{X} .

On note

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\mathbf{b} = (\mathbf{I} - \mathbf{H})\mathbf{y}$$

le vecteur des résidus ; c'est la projection de \mathbf{y} sur le sous-espace orthogonal de $\text{Vect}(\mathbf{X})$ dans \mathbb{R}^n .

3.2 Propriétés

Les estimateurs des M.C. b_0, b_1, \dots, b_p sont des estimateurs sans biais : $E(\mathbf{b}) = \boldsymbol{\beta}$, et, parmi les estimateurs sans biais fonctions linéaires des y_i , ils sont de variance minimum (théorème de Gauss-Markov) ; ils sont donc “BLUE” : *best linear unbiased estimators*. Sous hypothèse de normalité, les estimateurs du M.V. sont uniformément meilleurs (efficaces) et coïncident avec ceux des M.C.

On montre que la matrice de covariance des estimateurs se met sous la forme

$$E[(\mathbf{b} - \boldsymbol{\beta})(\mathbf{b} - \boldsymbol{\beta})'] = \sigma^2(\mathbf{X}'\mathbf{X})^{-1},$$

celle des prédictors est

$$E[(\hat{\mathbf{y}} - \mathbf{X}\boldsymbol{\beta})(\hat{\mathbf{y}} - \mathbf{X}\boldsymbol{\beta})'] = \sigma^2\mathbf{H}$$

et celle des estimateurs des résidus est

$$E[(\mathbf{e} - \mathbf{u})(\mathbf{e} - \mathbf{u})'] = \sigma^2(\mathbf{I} - \mathbf{H})$$

tandis qu'un estimateur sans biais de σ^2 est fourni par :

$$s^2 = \frac{\|\mathbf{e}\|^2}{n - p - 1} = \frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2}{n - p - 1} = \frac{\text{SSE}}{n - p - 1}.$$

Ainsi, les termes $s^2 h_i^2$ sont des estimations des variances des prédictors \hat{y}_i .

3.3 Sommes des carrés

SSE est la somme des carrés des résidus (*sum of squared errors*),

$$\text{SSE} = \|\mathbf{y} - \hat{\mathbf{y}}\|^2 = \|\mathbf{e}\|^2.$$

On définit également la somme totale des carrés (*total sum of squares*) par

$$\text{SST} = \|\mathbf{y} - \bar{y}\mathbf{1}\|^2 = \mathbf{y}'\mathbf{y} - n\bar{y}^2$$

et la somme des carrés de la régression (*regression sum of squares*) par

$$\text{SSR} = \|\hat{\mathbf{y}} - \bar{y}\mathbf{1}\|^2 = \hat{\mathbf{y}}'\hat{\mathbf{y}} - n\bar{y}^2 = \mathbf{y}'\mathbf{H}\mathbf{y} - n\bar{y}^2 = \mathbf{b}'\mathbf{X}'\mathbf{y} - n\bar{y}^2.$$

On vérifie alors : $\text{SST} = \text{SSR} + \text{SSE}$.

3.4 Coefficient de détermination

On appelle *coefficient de détermination* le rapport

$$R^2 = \frac{\text{SSR}}{\text{SST}}$$

qui est donc la part de variation de Y expliquée par le modèle de régression. Géométriquement, c'est un rapport de carrés de longueur de deux vecteurs. C'est donc le cosinus carré de l'angle entre ces vecteurs : \mathbf{y} et sa projection $\hat{\mathbf{y}}$ sur $\text{Vect}(\mathbf{X})$.

Attention, dans le cas extrême où $n = (p + 1)$, c'est-à-dire si le nombre de variables explicatives est grand comparativement au nombre d'observations, $R^2 = 1$. Ou encore, il est géométriquement facile de voir que l'ajout de variables explicatives ne peut que faire croître le coefficient de détermination.

La quantité R est appelée *coefficient de corrélation multiple* entre Y et les variables explicatives, c'est le coefficient de corrélation usuel entre \mathbf{y} et sa prédiction (ou projection) $\hat{\mathbf{y}}$.

4 Inférences dans le cas gaussien

En principe, l'hypothèse optionnelle (iv) de normalité des erreurs est nécessaire pour cette section. En pratique, des résultats asymptotiques, donc valides pour de grands échantillons, ainsi que des études de simulation, montrent que cette hypothèse n'est pas celle dont la violation est la plus pénalisante pour la fiabilité des modèles.

4.1 Inférence sur les coefficients

Pour chaque coefficient β_j on montre que la statistique

$$\frac{b_j - \beta_j}{\sigma_{b_j}}$$

où $\sigma_{b_j}^2$, variance de b_j est le j ème terme diagonal de la matrice $s^2(\mathbf{X}'\mathbf{X})^{-1}$, suit une loi de Student à $(n - p - 1)$ degrés de liberté. Cette statistique est donc utilisée pour tester

une hypothèse $H_0 : \beta_j = a$ ou pour construire un intervalle de confiance de niveau $100(1 - \alpha)\%$:

$$b_j \pm t_{\alpha/2; (n-p-1)} \sigma_{b_j}.$$

Attention, cette statistique concerne un coefficient et ne permet pas d'inférer conjointement (cf. §3.4) sur d'autres coefficients car ils sont corrélés entre eux ; de plus elle dépend des absences ou présences des autres variables X^k dans le modèle. Par exemple, dans le cas particulier de deux variables X^1 et X^2 très corrélées, chaque variable, en l'absence de l'autre, peut apparaître avec un coefficient significativement différent de 0 ; mais, si les deux sont présentes dans le modèle, elles peuvent chacune apparaître avec des coefficients insignifiants.

De façon plus générale, si \mathbf{c} désigne un vecteur non nul de $(p+1)$ constantes réelles, il est possible de tester la valeur d'une combinaison linéaire $\mathbf{c}'\mathbf{b}$ des paramètres en considérant l'hypothèse nulle $H_0 : \mathbf{c}'\mathbf{b} = a$; a connu. Sous H_0 , la statistique

$$\frac{\mathbf{c}'\mathbf{b} - a}{(s^2 \mathbf{c}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{c})^{1/2}}$$

suit une loi de Student à $(n - p - 1)$ degrés de liberté.

4.2 Inférence sur le modèle

Le modèle peut être testé globalement. Sous l'hypothèse nulle $H_0 : \beta_1 = \beta_2 = \dots = \beta_p = 0$, la statistique

$$\frac{\text{SSR}/p}{\text{SSE}/(n - p - 1)} = \frac{\text{MSR}}{\text{MSE}}$$

suit une loi de Fisher avec p et $(n - p - 1)$ degrés de liberté. Les résultats sont habituellement présentés dans un tableau "*d'analyse de la variance*" sous la forme suivante :

<i>Source de variation</i>	<i>d.d.l.</i>	<i>Somme des carrés</i>	<i>Variance</i>	<i>F</i>
Régression	p	SSR	$\text{MSR} = \text{SSR}/p$	MSR/MSE
Erreur	$n - p - 1$	SSE	$\text{MSE} = \text{SSE}/(n - p - 1)$	
Total	$n - 1$	SST		

4.3 Inférence sur un modèle réduit

Le test précédent amène à rejeter H_0 dès que l'une des variables X^j est liée à Y . Il est donc d'un intérêt limité. Il est souvent plus utile de tester un modèle réduit c'est-à-dire dans lequel certains coefficients, à l'exception de la constante, sont nuls contre le modèle complet avec toutes les variables. En ayant éventuellement réordonné les variables, on considère l'hypothèse nulle $H_0 : \beta_1 = \beta_2 = \dots = \beta_q = 0, q < p$.

Notons respectivement $\text{SSR}_q, \text{SSE}_q, R_q^2$ les sommes de carrés et le coefficient de détermination du modèle réduit à $(p - q)$ variables. Sous H_0 , la statistique

$$\frac{(\text{SSR} - \text{SSR}_q)/q}{\text{SSE}/(n - p - 1)} = \frac{(R^2 - R_q^2)/q}{(1 - R^2)/(n - p - 1)}$$

suit une loi de Fisher à q et $(n - p - 1)$ degrés de liberté.

Dans le cas particulier où $q = 1$ ($\beta_j = 0$), la F -statistique est alors le carré de la t -statistique de l'inférence sur un paramètre et conduit donc au même test.

4.4 Prédiction

Connaissant les valeurs des variables X^j pour une nouvelle observation : $\mathbf{x}'_0 = [x_0^1, x_0^2, \dots, x_0^p]$ appartenant au domaine dans lequel l'hypothèse de linéarité reste valide, une prédiction, notée \hat{y}_0 de Y ou $E(Y)$ est donnée par :

$$\hat{y}_0 = b_0 + b_1 x_0^1 + \dots + b_p x_0^p.$$

Les intervalles de confiance des prévisions de Y et $E(Y)$, pour une valeur $\mathbf{x}_0 \in \mathbb{R}^p$ et en posant $\mathbf{v}_0 = (1 | b\mathbf{m}x'_0)' \in \mathbb{R}^{p+1}$, sont respectivement

$$\begin{aligned} \hat{y}_0 \pm t_{\alpha/2; (n-p-1)} s(1 + \mathbf{v}'_0(\mathbf{X}'\mathbf{X})^{-1}\mathbf{v}_0)^{1/2}, \\ \hat{y}_0 \pm t_{\alpha/2; (n-p-1)} s(\mathbf{v}'_0(\mathbf{X}'\mathbf{X})^{-1}\mathbf{v}_0)^{1/2}. \end{aligned}$$

4.5 Exemple

Le modèle de régression linéaire n'est pas adapté à l'explication d'une variable binaire comme dans le cas des données bancaires. Ceci est abordé dans le chapitre suivant en utilisant la régression logistique tandis que d'autres exemples de données sont utilisées dans ce chapitre. Les premières sont extraites de Jobson (1991) et décrivent les résultats comptables de 40 entreprises du Royaume Uni.

RETCAP	Return on capital employed
WCFTDT	Ratio of working capital flow to total debt
LOGSALE	Log to base 10 of total sales
LOGASST	Log to base 10 of total assets
CURRAT	Current ratio
QUIKRAT	Quick ratio
NFATAST	Ratio of net fixed assets to total assets
FATTOT	Gross fixed assets to total assets
PAYOUT	Payout ratio
WCFTCL	Ratio of working capital flow to total current liabilities
GEARRAT	Gearing ratio (debt-equity ratio)
CAPINT	Capital intensity (ratio of total sales to total assets)
INVTAST	Ratio of total inventories to total assets

Modèle complet

La procédure SAS/REG est utilisée dans le programme suivant. Beaucoup d'options sont actives afin de fournir la plupart des résultats même si certains sont redondants ou peu utiles.

```
options linesize=110 pagesize=30 nodate nonumber;
title;
proc reg data=sasuser.ukcomp1 all;
```

```

model RETCAP = WCFTCL  WCFTDT  GEARRAT  LOGSALE  LOGASST
              NFATAST  CAPINT  FATTOT  INVTAST  PAYOUT  QUIKRAT  CURRAT
              /dw covb  Influence cli clm tol vif collin R P;
output out=resout h=lev p=pred r=res student=resstu ;
run;

```

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Prob>F
	(1)				
Model	12	0.55868 (2)	0.04656 (5)	8.408 (7)	0.0001 (8)
Error	27	0.14951 (3)	0.00554 (6)		
C Total	39	0.70820 (4)			
Root MSE	0.07441 (9)	R-square	0.7889 (12)		
Dep Mean	0.14275 (10)	Adj R-sq	0.6951 (13)		
C.V.	52.12940 (11)				

-
- (1) degrés de liberté de la loi de Fisher du test global
 - (2) SSR
 - (3) SSE ou déviance
 - (4) SST=SSE+SSR
 - (5) SSR/DF
 - (6) $s^2 = \text{MSE} = \text{SSE}/\text{DF}$ est l'estimation de σ^2
 - (7) Statistique F du test de Fisher du modèle global
 - (8) $P(f_{p;n-p-1} > F)$; H_0 est rejetée au niveau α si $P < \alpha$
 - (9) s = racine de MSE
 - (10) moyenne empirique de la variable à expliquée
 - (11) Coefficient de variation $100 \times (9)/(10)$
 - (12) Coefficient de détermination R^2
 - (13) Coefficient de détermination ajusté R'^2
-

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob> T	Tolerance	Variance Inflation
		(1)	(2)	(3)	(4)	(5)	(6)
INTERCEP	1	0.188072	0.13391661	1.404	0.1716	.	0.00000000
WCFTCL	1	0.215130	0.19788455	1.087	0.2866	0.03734409	26.77799793
WCFTDT	1	0.305557	0.29736579	1.028	0.3133	0.02187972	45.70441500
GEARRAT	1	-0.040436	0.07677092	-0.527	0.6027	0.45778579	2.18442778
LOGSALE	1	0.118440	0.03611612	3.279	0.0029	0.10629382	9.40788501
LOGASST	1	-0.076960	0.04517414	-1.704	0.0999	0.21200778	4.71680805

...

-
- (1) estimations des paramètres (b_j)
 - (2) écarts-types de ces estimations (s_{b_j})
 - (3) statistique T du test de Student de $H_0 : b_j = 0$
 - (4) $P(t_{n-p-1} > T)$; H_0 est rejetée au niveau α si $P < \alpha$
 - (5) $1 - R_{(j)}^2$
 - (6) $\text{VIF} = 1/(1 - R_{(j)}^2)$
-

Ces résultats soulignent les problèmes de colinéarités. De grands “VIF” sont associés à de grands écart-types des estimations des paramètres. D’autre part les nombreux tests de Student non significatifs montrent que trop de variables sont présentes dans le modèle. Cette idée est renforcée par le calcul de l’indice de conditionnement (explicité dans la section suivante : 8.76623/0.00125).

5 Choix de modèle

De façon un peu schématique, on peut associer la pratique de la modélisation statistique à trois objectifs qui peuvent éventuellement être poursuivis en complémentarité.

Descriptif : Il vise à rechercher de façon exploratoire les liaisons entre Y et d'autres variables, potentiellement explicatives, X^j qui peuvent être nombreuses afin, par exemple d'en sélectionner un sous-ensemble. À cette stratégie, à laquelle peuvent contribuer des Analyses en Composantes Principales, correspond des algorithmes de recherche (pas à pas) moins performants mais économiques en temps de calcul si p est grand.

Attention, si n est petit, et la recherche suffisamment longue avec beaucoup de variables explicatives, il sera toujours possible de trouver un “bon” modèle expliquant y ; c'est l'effet *data mining* dans les modèles économétriques appelé maintenant *data snooping*.

Explicatif : Le deuxième objectif est sous-tendu par une connaissance *a priori* du domaine concerné et dont des résultats théoriques peuvent vouloir être confirmés, infirmés ou précisés par l'estimation des paramètres. Dans ce cas, les résultats inférentiels précédents permettent de construire le bon test conduisant à la prise de décision recherchée. Utilisées hors de ce contexte, les statistiques de test n'ont plus alors qu'une valeur indicative au même titre que d'autres critères plus empiriques.

Prédicatif : Dans le troisième cas, l'accent est mis sur la qualité des estimateurs et des prédicteurs qui doivent, par exemple, minimiser une erreur quadratique moyenne. C'est la situation rencontrée en *apprentissage*. Ceci conduit à rechercher des modèles *parcimonieux* c'est-à-dire avec un nombre volontairement restreint de variables explicatives. Le “meilleur” modèle ainsi obtenu peut donner des estimateurs légèrement biaisés au profit d'un compromis pour une variance plus faible. Un bon modèle n'est donc plus celui qui explique le mieux les données au sens d'une déviance (SSE) minimale (ou d'un R^2 max) au prix d'un nombre important de variables pouvant introduire des colinéarités. Le bon modèle est celui qui conduit aux prédictions les plus fiables.

Certes, le théorème de Gauss-Markov indique que, parmi les estimateurs sans biais, celui des moindres carrés est de variance minimum. Néanmoins, il peut être important de préférer un estimateur légèrement biaisé si le gain en variance est lui plus significatif. C'est tout le problème de trouver un bon équilibre entre biais et variance afin de minimiser un risque quadratique de prédiction. Il y a principalement deux façons de “biaiser” un modèle dans le but de restreindre la variance :

- en réduisant le nombre de variables explicatives et donc en simplifiant le modèle,
- en contraignant les paramètres du modèle, en les rétrécissant (*shrinkage*), en régression *ridge* qui opère une régularisation.

Commençons par décrire les procédures de sélection.

5.1 Critères

De nombreux critères de choix de modèle sont présentés dans la littérature sur la régression linéaire multiple. Citons le critère d'information d'Akaike (AIC), celui bayésien de Sawa (BIC)... (cf. chapitre 3). Ils sont équivalents lorsque le nombre de variables à sélectionner, ou niveau du modèle, est fixé. Le choix du critère est déterminant lorsqu'il

s'agit de comparer des modèles de niveaux différents. Certains critères se ramènent, dans le cas gaussien, à l'utilisation d'une expression pénalisée de la fonction de vraisemblance afin de favoriser des modèles parcimonieux. En pratique, les plus utilisés ou ceux généralement fournis par les logiciels sont les suivants.

Statistique du F de Fisher

Ce critère, justifié dans le cas explicatif car basé sur une qualité d'ajustement est aussi utilisé à titre indicatif pour comparer des séquences de modèles emboîtés. La statistique partielle de Fisher est

$$\frac{(\text{SSR} - \text{SSR}_q)/s}{\text{SSE}/(n-p-1)} = \frac{(R^2 - R_q^2)}{1 - R^2} \frac{n-p-1}{q}$$

dans laquelle l'indice q désigne les expressions concernant le modèle réduit avec $(p-q)$ variables explicatives. On considère alors que si l'accroissement $(R^2 - R_q^2)$ est suffisamment grand :

$$R^2 - R_q^2 > \frac{q}{(n-p-1)} F_{\alpha; q, (n-p-1)},$$

l'ajout des q variables au modèle est justifié.

R^2 et R^2 ajusté

Le coefficient de détermination $R^2 = 1 - \text{SSE}/\text{SST}$, directement lié à la déviance (SSE) est aussi un indice de qualité mais qui a la propriété d'être monotone croissant en fonction du nombre de variables. Il ne peut donc servir qu'à comparer deux modèles de même niveau c'est-à-dire avec le même nombre de variables.

En revanche, le R^2 ajusté :

$$R'^2 = 1 - \frac{n-1}{n-p-1} (1 - R^2) = 1 - \frac{\text{SSE}/(n-p-1)}{\text{SST}/(n-1)}.$$

dans lequel le rapport SSE/SST est remplacé par un rapport des estimations sans biais des quantités σ^2 et σ_y^2 introduit une pénalisation liée au nombre de paramètres à estimer.

Ce coefficient s'exprime encore par

$$1 - \frac{(n-1)\text{MSE}}{\text{SST}}$$

ainsi dans la comparaison de deux modèles partageant la même SST, on observe que $R'^2 > R'_j{}^2$ si et seulement si $\text{MSE} < \text{MSE}_j$; MSE et MSE_j désignant respectivement l'erreur quadratique moyenne du modèle complet et celle d'un modèle à j variables explicatives. Maximiser le R^2 ajusté revient donc à minimiser l'erreur quadratique moyenne.

C_p de Mallows

Cet indicateur est une estimation de l'erreur quadratique moyenne de prédiction qui s'écrit aussi comme la somme d'une variance et du carré d'un biais. L'erreur quadratique moyenne de prédiction s'écrit ainsi :

$$\text{MSE}(\hat{y}_i) = \text{Var}(\hat{y}_i) + [\text{Biais}(\hat{y}_i)]^2$$

puis après sommation et réduction :

$$\frac{1}{\sigma^2} \sum_{i=1}^n \text{MSE}(\hat{y}_i) = \frac{1}{\sigma^2} \sum_{i=1}^n \text{Var}(\hat{y}_i) + \frac{1}{\sigma^2} \sum_{i=1}^n [\text{Biais}(\hat{y}_i)]^2.$$

En supposant que les estimations du modèle complet sont sans biais et en utilisant des estimateurs de $\text{Var}(\hat{y}_i)$ et σ^2 , l'expression de l'erreur quadratique moyenne totale standardisée (ou réduite) pour un modèle à j variables explicatives s'écrit :

$$C_p = (n - q - 1) \frac{\text{MSE}_j}{\text{MSE}} - [n - 2(q + 1)]$$

et définit la valeur du C_p de Mallows pour les q variables considérées. Il est alors d'usage de rechercher un modèle qui minimise le C_p tout en fournissant une valeur inférieure et proche de $(q + 1)$. Ceci revient à considérer que le “vrai” modèle complet est moins fiable qu'un modèle réduit donc biaisé mais d'estimation plus précise.

Akaike's Information criterion (AIC)

A compléter

PRESS de Allen

Il s'agit l'introduction historique de la validation croisée. On désigne par $\hat{y}_{(i)}$ la prédiction de y_i calculée sans tenir compte de la i ème observation $(y_i, x_i^1, \dots, x_i^p)$, la somme des erreurs quadratiques de prédiction (PRESS) est définie par

$$\text{PRESS} = \sum_{i=1}^n (y_i - \hat{y}_{(i)})^2$$

et permet de comparer les capacités prédictives de deux modèles. Le chapitre 3 donne plus de détails sur ce type d'estimation.

5.2 Algorithmes de sélection

Lorsque p est grand, il n'est pas raisonnable de penser explorer les 2^p modèles possibles afin de sélectionner le “meilleur” au sens de l'un des critères ci-dessus. Différentes stratégies sont donc proposées qui doivent être choisies en fonction de l'objectif recherché et des moyens de calcul disponibles ! Trois types d'algorithmes sont résumés ci-dessous par ordre croissant de temps de calcul nécessaire c'est-à-dire par nombre croissant de modèles considérés parmi les 2^p et donc par capacité croissante d'optimalité. On donne pour chaque algorithme l'option `selection` à utiliser dans la procédure `REG` de SAS.

Pas à pas

Sélection (forward) À chaque pas, une variable est ajoutée au modèle. C'est celle dont la valeur p (“prob value”) associée à la statistique partielle du test de Fisher qui compare les deux modèles est minimum. La procédure s'arrête lorsque toutes les variables sont introduites ou lorsque p reste plus grande qu'une valeur seuil fixée par défaut à 0,50.

Élimination (backward) L'algorithme démarre cette fois du modèle complet. À chaque étape, la variable associée à la plus grande valeur p est éliminée du modèle. La procédure s'arrête lorsque les variables restant dans le modèle ont des valeurs p plus petites qu'un seuil fixé par défaut à 0,10.

Mixte (stepwise) Cet algorithme introduit une étape d'élimination de variable après chaque étape de sélection afin de retirer du modèle d'éventuels variables qui seraient devenues moins indispensables du fait de la présence de celles nouvellement introduites.

Global

L'algorithme de Furnival et Wilson est utilisé pour comparer tous les modèles possibles en cherchant à optimiser l'un des critères : R^2 , R^2 ajusté, ou C_p de Mallows (**rsquare**, **adjrsq**, **cp**) choisi par l'utilisateur. Par souci d'économie, cet algorithme évite de considérer des modèles de certaines sous-branches de l'arborescence dont on peut savoir a priori qu'ils ne sont pas compétitifs. En général les logiciels exécutant cet algorithme affichent le (**best=1**) ou les meilleurs modèles de chaque niveau.

5.3 Exemple

Parmi les trois types d'algorithmes et les différents critères de choix, une des façons les plus efficaces consistent à choisir les options du programme ci-dessous. Tous les modèles (parmi les plus intéressants selon l'algorithme de Furnival et Wilson) sont considérés. Seul le meilleur pour chaque niveau, c'est-à-dire pour chaque valeur p du nombre de variables explicatives sont donnés. Il est alors facile de choisir celui minimisant l'un des critères globaux (C_p ou BIC ou ...).

```
options linesize=110 pagesize=30 nodate nonumber;
title;
```

```
proc reg data=sasuser.ukcomp2 ;
model RETCAP = WCFTCL WCFTDT GEARRAT LOGSALE LOGASST
              NFATAST CAPINT FATTOT INVTAST PAYOUT QUIKRAT CURRAT
              / selection=rsquare cp rsquare bic best=1;
run;
```

```

      N = 40      Regression Models for Dependent Variable: RETCAP
R-square Adjusted C(p)      BIC      Variables in Model
In      R-square
1  0.1055 0.0819 78.3930 -163.26 WCFTCL
2  0.3406 0.3050 50.3232 -173.72 WCFTDT QUIKRAT
3  0.6154 0.5833 17.1815 -191.14 WCFTCL NFATAST CURRAT
4  0.7207 0.6888  5.7146 -199.20 WCFTDT LOGSALE NFATAST CURRAT
5  0.7317 0.6923  6.3047 -198.05 WCFTDT LOGSALE NFATAST QUIKRAT CURRAT
6  0.7483 0.7025  6.1878 -197.25 WCFTDT LOGSALE NFATAST INVTAST QUIKRAT CURRAT
7  0.7600 0.7075  6.6916 -195.77 WCFTDT LOGSALE LOGASST NFATAST FATTOT QUIKRAT CURRAT
8  0.7692 0.7097  7.5072 -193.87 WCFTDT LOGSALE LOGASST NFATAST FATTOT INVTAST QUIKRAT CURRAT
9  0.7760 0.7088  8.6415 -191.59 WCFTCL WCFTDT LOGSALE LOGASST NFATAST FATTOT INVTAST QUIKRAT
    CURRAT
10 0.7830 0.7082  9.7448 -189.15 WCFTCL WCFTDT LOGSALE LOGASST NFATAST FATTOT INVTAST PAYOUT
    QUIKRAT CURRAT
11 0.7867 0.7029 11.2774 -186.40 WCFTCL WCFTDT LOGSALE LOGASST NFATAST CAPINT FATTOT INVTAST
    PAYOUT QUIKRAT CURRAT
```

```
12  0.7888 0.6950 13.0000 -183.51 WCFTCL WCFTDT GEARRAT LOGSALE LOGASST NFATAST CAPINT FATTOT
      INVTAST PAYOUT QUIKRAT CURRAT
```

Dans cet exemple, C_p et BIC se comportent de la même façon. Avec peu de variables, le modèle est trop biaisé. Ils atteignent un minimum pour un modèle à 4 variables explicatives puis croissent de nouveau selon la première bissectrice. La maximisation du R^2 ajusté conduirait à une solution beaucoup moins parcimonieuse. On note par ailleurs que l'algorithme remplace WCFTCL par WCFTDT. Un algorithme par sélection ne peut pas aboutir à la solution optimale retenue.

5.4 Choix de modèle par régularisation

L'autre stratégie qui cherche à conserver l'ensemble ou tout du moins la plupart des variables explicatives pose un problème de *multicolinéarité*. Il est résolu par une procédure de régularisation.

Problème

L'estimation des paramètres ainsi que celle de leur écart-type (standard error) nécessite le calcul explicite de la matrice $(\mathbf{X}'\mathbf{X})^{-1}$. Dans le cas dit *mal conditionné* où le déterminant de la matrice $\mathbf{X}'\mathbf{X}$ n'est que légèrement différent de 0, les résultats conduiront à des estimateurs de variances importantes et même, éventuellement, à des problèmes de précision numérique. Il s'agit donc de diagnostiquer ces situations critiques puis d'y remédier. Dans les cas descriptif ou prédictif on supprime des variables à l'aide des procédures de choix de modèle mais, pour un objectif explicatif nécessitant toutes les variables, d'autres solutions doivent être envisagées : algorithme de résolution des équations normales par transformations orthogonales (procédure **orthoreg** de SAS) sans calcul explicite de l'inverse pour limiter les problèmes numériques, régression biaisée (ridge), régression sur composantes principales.

VIF

La plupart des logiciels proposent des diagnostics de colinéarité. Le plus classique est le *facteur d'inflation de la variance* (VIF)

$$V_j = \frac{1}{1 - R_j^2}$$

où R_j^2 désigne le coefficient de détermination de la régression de la variable X^j sur les autres variables explicatives ; R_j est alors un coefficient de corrélation multiple, c'est le cosinus de l'angle dans \mathbb{R}^n entre X^j et le sous-espace vectoriel engendré par les variables $\{X^1, \dots, X^{j-1}, X^{j+1}, \dots, X^p\}$. Plus X^j est "linéairement" proche de ces variables et plus R_j est proche de 1 ; on montre alors que la variance de l'estimateur de β_j est d'autant plus élevée. Évidemment, cette variance est minimum lorsque X^j est orthogonale au sous-espace engendré par les autres variables.

Conditionnement

De façon classique, les qualités numériques de l'inversion d'une matrice sont quantifiées par son *indice de conditionnement*. On note $\lambda_1, \dots, \lambda_p$ les valeurs propres de la matrice des corrélations \mathbf{R} rangées par ordre décroissant. Le déterminant de \mathbf{R} est égal au produit des valeurs propres. Ainsi, des problèmes numériques, ou de variances excessives apparaissent dès que les dernières valeurs propres sont relativement trop petites. L'*indice de conditionnement* est le rapport

$$\kappa = \lambda_1 / \lambda_p$$

de la plus grande sur la plus petite valeur propre.

En pratique, si $\kappa < 100$ on considère qu'il n'y a pas de problème. Celui-ci devient sévère pour $\kappa > 1000$. Cet indice de conditionnement donne un aperçu global des problèmes de colinéarité tandis que les VIF, les tolérances ou encore l'étude des vecteurs propres associés au plus petites valeurs propres permettent d'identifier les variables les plus problématiques.

Régression *ridge*

Ayant diagnostiqué un problème mal conditionné mais désirant conserver toutes les variables, il est possible d'améliorer les propriétés numériques et la variance des estimations en considérant un estimateur légèrement biaisé des paramètres. L'estimateur "*ridge*" est donné par

$$\mathbf{b}_R = (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}'\mathbf{y},$$

qui a pour effet de décaler de la valeur k toutes les valeurs propres de la matrice à inverser et, plus particulièrement, les plus petites qui reflètent la colinéarité. On montre que cela revient encore à estimer le modèle par les moindres carrés sous la contrainte que la norme du vecteur¹ $\boldsymbol{\beta}$ des paramètres ne soit pas trop grande :

$$\mathbf{b}_R = \arg \min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 ; \|\boldsymbol{\beta}\|^2 < c \right\}.$$

C'est encore, en introduisant un multiplicateur de Lagrange dans le problème de minimisation, un problème de moindres carrés pénalisés :

$$\mathbf{b}_R = \arg \min_{\boldsymbol{\beta}} \{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|^2 \}.$$

Cela revient à pénaliser la norme de l'estimateur pour empêcher les coefficients d'exploser et donc pour limiter la variance. On parle aussi d'estimateur à rétrécisseur (*shrinkage*). Comme dans tout problème de régularisation, il est nécessaire de fixer la valeur du paramètre λ ; la validation croisée peut être utilisée à cette fin mais la lecture du graphique (cf. figure 2.1) montrant l'évolution des paramètres en fonction du coefficient ridge est souvent suffisante. La valeur est choisie au point où la décroissance des paramètres devient faible et quasi-linéaire. Une autre version (*lasso*) de régression biaisée est obtenue en utilisant la norme en valeur absolue pour définir la contrainte sur les paramètres.

¹En pratique, la contrainte ne s'applique pas au terme constant β_0 mais seulement aux coefficients du modèle.

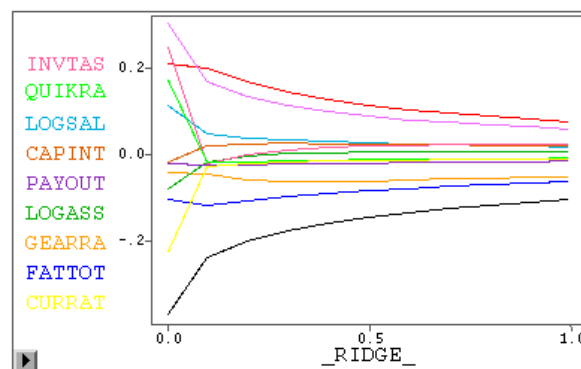


FIG. 2.1 – Evolution des paramètres de la régression ridge en fonction du paramètre de régularisation.

Régression sur composantes principales

L'Analyse en Composantes Principales est, entre autres, la recherche de p variables dites principales qui sont des combinaisons linéaires des variables initiales de variance maximale sous une contrainte d'orthogonalité (cf. Baccini et Besse (2000) pour des détails). En désignant par \mathbf{V} la matrice des vecteurs propres de la matrice des corrélations \mathbf{R} rangés dans l'ordre décroissant des valeurs propres, les valeurs prises par ces variables principales sont obtenues dans la matrice des composantes principales

$$\mathbf{C} = (\mathbf{X} - \mathbf{1}\bar{\mathbf{x}}')\mathbf{V}.$$

Elles ont chacune pour variance la valeur propre λ_j associée. Le sous-espace engendré par ces variables principales est le même que celui engendré par les variables initiales. Il est donc géométriquement équivalent de régresser Y sur les colonnes de \mathbf{C} que sur celles de \mathbf{X} . Les problèmes de colinéarité sont alors résolus en supprimant les variables principales de plus faibles variances c'est-à-dire associées aux plus petites valeurs propres ou encore en exécutant un algorithme de choix de modèle sur les composantes.

La solution obtenue présente ainsi de meilleures qualités prédictives mais, les coefficients de la régression s'appliquant aux composantes principales, un calcul complémentaire est nécessaire afin d'évaluer et d'interpréter les effets de chacune des variables initiales.

Régression PLS

Une dernière approche est largement utilisée afin de pouvoir traiter les situations avec une forte multicolinéarité et même, lorsque le nombre d'observations est inférieur au nombre de prédicteurs. Il s'agit de la régression PLS (*partial least square*). Comme pour la régression sur composantes principales, celle-ci est décomposée sur une base orthogonale construite à partir de combinaisons linéaires des variables explicatives centrées réduites mais la construction de cette base dépend de la corrélation des prédicteurs avec Y . Il s'agit d'une démarche itérative. À chaque étape, est recherchée la combinaison linéaire orthogonale aux solutions précédentes et la plus liée à la variable à expliquer. La première étape est obtenue par la régression de Y sur chacune des variables explicatives.

- **Initialisation** Les variables X^j sont centrées et réduites, on pose $\hat{\mathbf{y}}^{(0)} = \mathbf{1}\bar{y}$ et $\mathbf{x}_j^{(0)} = \mathbf{x}_j; j = 1, \dots, p$.
 - **Pour $m = 1$ à p Faire**
 - $\mathbf{z}_m = \sum_{j=1}^p \alpha_{mj} \mathbf{x}_j^{(m-1)}$; avec $\alpha_{mj} = \langle \mathbf{x}_j^{(m-1)}, \mathbf{y} \rangle$.
 - $\theta_m = \langle \mathbf{z}_m, \mathbf{y} \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle$.
 - $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \theta_m \mathbf{z}_m$.
 - **Orthogonalisation** : $\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - \left[\langle \mathbf{x}_j^{(m-1)}, \mathbf{z}_m \rangle / \langle \mathbf{z}_m, \mathbf{z}_m \rangle \right] \mathbf{z}_m ; j = 1, \dots, p$.
 - **Fin pour**
 - Le résultat est $\hat{\mathbf{y}}^{(q)}$ après un choix de $m = q$ composantes. Les coefficients sur les variables explicatives initiales sont donnés par : $\beta_j^{pls} = \sum_{l=1}^q \alpha_{lj} \theta_l$.
-

6 Compléments

6.1 Modèles curvilinéaires

En cas d'invalidation de l'hypothèse de linéarité, il peut être intéressant de considérer des modèles polynômiaux, très classiques pour décrire des phénomènes physiques, de la forme

$$Y = \beta_0 + \dots + \beta_j X^j + \dots + \gamma_{kl} X^k X^l + \dots + \delta_j X^{j^2}$$

qui sont encore appelés *surfaces de réponse* en planification expérimentale. Ces modèles sont faciles à étudier dans le cadre linéaire, il suffit d'ajouter des nouvelles variables constituées des produits ou des carrés des variables explicatives initiales. Les choix : présence ou non d'une interaction entre deux variables, présence ou non d'un terme quadratique se traitent alors avec les mêmes outils que ceux des choix de variable mais en intégrant une contrainte lors de la lecture des résultats : ne pas considérer des modèles incluant des termes quadratiques dont les composants linéaires auraient été exclus ou encore, ne pas supprimer d'un modèle une variable d'un effet linéaire si elle intervient dans un terme quadratique.

La procédure **rsreg** de SAS est plus particulièrement adaptée aux modèles quadratiques. Elle ne comporte pas de procédure de choix de modèle mais fournit des aides et diagnostics sur l'ajustement de la surface ainsi que sur la recherche des points optimaux.

Attention : Ce type de modèle accroît considérablement les risques de colinéarité, il est peu recommandé de considérer des termes cubiques.

6.2 Influence, résidus, validation

Avant toute tentative de modélisation complexe, il est impératif d'avoir conduit des analyses uni et bivariées afin d'identifier des problèmes sur les distributions de chacune des variables : dissymétrie, valeurs atypiques (outliers) ou sur les liaisons des variables prises deux par deux : non-linéarité. Ces préliminaires acquis, des aides ou diagnostics associés à la régression linéaire multiple permettent de détecter des violations d'hypothèses (homoscédasticité, linéarité) ou des points influents dans ce contexte multidimensionnel (cf. figure 2.2).

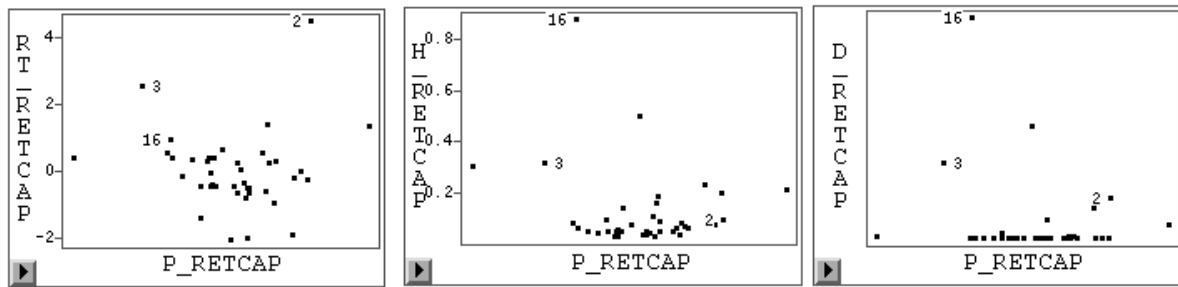


FIG. 2.2 – Graphe des résidus studentisés, de la diagonale de la matrice \mathbf{H} et de la distance de Cook en fonction des valeurs prédites.

(1)	variable à expliquer y_i
(2)	valeur ajustée \hat{y}_i
(3)	écart-type de cette estimation $s_{\hat{y}_i}$
(4) et (5)	Intervalle de confiance pour l'estimation de $E(y_i)$
(6) et (7)	Intervalle de confiance pour l'estimation de y_i
(8)	résidus calculés e_i
(9)	écarts-types de ces estimations
(10)	résidus standardisés (ou studentisés internes) r_i
(11)	repérage graphique des résidus standardisés : * = 0.5.
(12)	Distance de Cook
(13)	résidus studentisés (externes) t_i
(14)	Termes diagonaux de la matrice chapeau \mathbf{H}
(15)	autres indicateurs d'influence

```

Sum of Residuals           0
Sum of Squared Residuals   0.1495  (SSE)
Predicted Resid SS (Press) 1.0190  (PRESS)

```

Régression partielle

Un modèle de régression multiple est une technique *linéaire*. Il est raisonnable de s'interroger sur la pertinence du caractère linéaire de la contribution d'une variable explicative à l'ajustement du modèle. Ceci peut être réalisé en considérant une *régression partielle*.

On calcule alors deux régressions :

- la régression de Y sur les variables $X^1, \dots, X^{j-1}, X^{j+1}, \dots, X^p$, dans laquelle la j ème variable est omise, soit $\mathbf{r}_{y(j)}$ le vecteur des résidus obtenus.
- La régression de X^j sur les variables $X^1, \dots, X^{j-1}, X^{j+1}, \dots, X^p$. Soit $\mathbf{r}_{x(j)}$ le vecteur des résidus obtenus.

La comparaison des résidus par un graphe (nuage de points $\mathbf{r}_{y(j)} \times \mathbf{r}_{x(j)}$) permet alors de représenter la nature de la liaison entre X^j et Y *conditionnellement* aux autres variables explicatives du modèle.

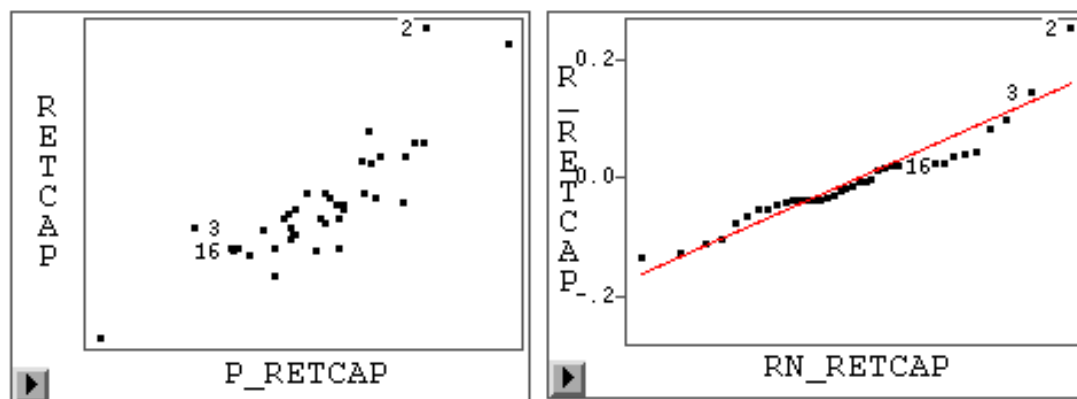


FIG. 2.3 – Graphe des valeurs observées en fonction des valeurs prédites et droite de Henri des résidus (normal qq-plot).

Graphes

Différents graphiques permettent finalement de contrôler le bien fondé des hypothèses de linéarité, d'homoscédasticité, éventuellement de normalité des résidus.

- Le premier considère le nuage de points des résidus studentisés croisés avec les valeurs prédites. Les points doivent être uniformément répartis entre les bornes -2 et $+2$ et ne pas présenter de formes suspectes (cf. figure 2.2).
- Le deuxième croise les valeurs observées de Y avec les valeurs prédites. Il illustre le coefficient de détermination R qui est aussi la corrélation linéaire simple entre \hat{y} et y . Les points doivent s'aligner autour de la première bissectrice. Il peut être complété par l'intervalle de confiance des y_i ou celui de leurs moyennes. (cf. figure 2.3).
- La qualité, en terme de linéarité, de l'apport de chaque variable est étudiée par des régressions partielles. Chaque graphe de résidus peut être complété par une estimation fonctionnelle ou régression non-paramétrique (loess, noyau, spline) afin d'en faciliter la lecture.
- Le dernier trace la droite de Henri (Normal QQplot) des résidus dont le caractère linéaire de la représentation donne une idée de la normalité de la distribution. (cf. figure 2.3)

7 Analyse de variance à un facteur

7.1 Introduction

Les techniques dites d'*analyse de variance* sont des outils entrant dans le cadre général du modèle linéaire et où une variable quantitative est expliquée par une ou plusieurs variables qualitatives. L'objectif essentiel est alors de comparer les moyennes empiriques de la variable quantitative observées pour différentes catégories d'unités statistiques. Ces catégories sont définies par l'observation des variables qualitatives ou *facteurs* prenant différentes modalités ou encore de variables quantitatives découpées en classes ou *niveaux*.

Une combinaison de niveaux définit une *cellule*, *groupe* ou *traitement*.

Il s'agit donc de savoir si un facteur ou une combinaison de facteurs (*interaction*) a un *effet* sur la variable quantitative en vue, par exemple, de déterminer des conditions optimales de production ou de fabrication, une dose optimale de médicaments. . . . Ces techniques apparaissent aussi comme des cas particuliers de la régression linéaire multiple en associant à chaque modalité une *variable indicatrice* (dummy variable) et en cherchant à expliquer une variable quantitative par ces variables indicatrices. L'appellation "analyse de variance" vient de ce que les tests statistiques sont bâtis sur des comparaisons de sommes de carrés de variations.

L'analyse de variance est souvent utilisée pour analyser des données issue d'une *planification expérimentale* au cours de laquelle l'expérimentateur a la possibilité de contrôler *a priori* les niveaux des facteurs avec pour objectif d'obtenir le maximum de précision au moindre coût. Ceci conduit en particulier à construire des facteurs orthogonaux deux à deux (variables explicatives non linéairement corrélées) afin de minimiser la variance des estimateurs. On distingue le cas particulier important où les cellules ont le même effectif, on parle alors de *plan orthogonal* ou *équiréparté* ou *équilibré* (balanced), qui conduit à des simplifications importantes de l'analyse de variance associée. On appelle *plan complet* un dispositif dans lequel toutes les combinaisons de niveaux ont été expérimentées. On distingue entre des modèles fixes, aléatoires ou mixtes selon le caractère déterministe (contrôlé) ou non des facteurs par exemple si les modalités résultent d'un choix aléatoire parmi un grand nombre de possibles. Dans cette courte introduction seuls le modèle fixe à un facteur est considéré.

L'analyse de variance à un facteur est un cas particulier d'étude de relations entre deux variables statistiques : une quantitative Y admettant une densité et une qualitative X ou facteur qui engendre une partition ou classification de l'échantillon en J groupes, cellules ou classes indicées par j . L'objectif est de comparer les distributions de Y pour chacune des classes en particulier les valeurs des moyennes et variances. Un préalable descriptif consiste à réaliser un graphique constitué de diagrammes boîtes parallèles : une pour chaque modalité. Cette représentation donne une première appréciation de la comparaison des distributions (moyenne, variance) internes à chaque groupe. Les spécificités de la planification d'expérience ne sont pas abordées dans ce cours axé sur la fouille de données pour laquelle les données sont justement préalablement fournies. Les plans d'expérience sont surtout utilisés en milieu industriel : contrôle de qualité, optimisation des processus de production, ou en agronomie pour la sélection de variétés, la comparaison d'engrais, d'insecticides. . . . La bibliographie est abondante sur ce sujet.

7.2 Modèle

Pour chaque niveau j de X , on observe n_j valeurs $y_{1j}, \dots, y_{n_j j}$ de la variable Y et où $n = \sum_{j=1}^J n_j$ ($n > J$) est la taille de l'échantillon. On suppose qu'à l'intérieur de chaque cellule, les observations sont indépendantes équidistribuées de moyenne μ_j et de variance homogène $\sigma_j^2 = \sigma^2$. Ceci s'écrit :

$$y_{ij} = \mu_j + \varepsilon_{ij}$$

où les ε_{ij} sont i.i.d. suivant une loi centrée de variance σ^2 qui sera supposée $\mathcal{N}(0, \sigma^2)$ pour la construction des tests. Cette dernière hypothèse n'étant pas la plus sensible. Les espérances μ_j ainsi que le paramètre de nuisance σ^2 sont les paramètres inconnus à estimer.

On note respectivement :

$$\begin{aligned}\bar{y}_{.j} &= \frac{1}{n_j} \sum_{i=1}^{n_j} y_{ij}, \\ s_j^2 &= \frac{1}{n_j - 1} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{.j})^2, \\ \bar{y}_{..} &= \frac{1}{n} \sum_{i=1}^{n_j} \sum_{j=1}^J y_{ij},\end{aligned}$$

les moyennes et variances empiriques de chaque cellule, la moyenne générale de l'échantillon.

Les paramètres μ_j sont estimés sans biais par les moyennes $\bar{y}_{.j}$ et comme le modèle s'écrit alors :

$$y_{ij} = \bar{y}_{.j} + (y_{ij} - \bar{y}_{.j}),$$

l'estimation des erreurs est $e_{ij} = (y_{ij} - \bar{y}_{.j})$ tandis que les valeurs prédites sont $\hat{y}_{ij} = \bar{y}_{.j}$.

Sous l'hypothèse d'homogénéité des variances, la meilleure estimation sans biais de σ^2 est

$$s^2 = \frac{\sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{.j})^2}{n - J} = \frac{1}{n - J} [(n - 1)s_1^2 + \dots + (n_J - 1)s_J^2]$$

qui s'écrit donc comme une moyenne pondérée des variances empiriques de chaque groupe.

Notons \mathbf{y} le vecteur des observations $[y_{ij}|i = 1, n_j; j = 1, J]'$ mis en colonne, $\boldsymbol{\varepsilon} = [\varepsilon_{ij}|i = 1, n_j; j = 1, J]'$ le vecteur des erreurs, $\mathbf{1}_j$ les variables indicatrices des niveaux et $\mathbf{1}$ la colonne de 1s. Le i ème élément d'une variable indicatrice (dummy variable) $\mathbf{1}_j$ prend la valeur 1 si la i ème observation y_i est associée au j ème et 0 sinon.

Comme dans le cas de la régression linéaire multiple, le modèle consiste à écrire que l'espérance de la variable Y appartient au sous-espace linéaire engendré par les variables explicatives, ici les variables indicatrices :

$$\mathbf{y} = \beta_0 \mathbf{1} + \beta_1 \mathbf{1}_1 + \dots + \beta_J \mathbf{1}_J + \boldsymbol{\varepsilon}.$$

La matrice \mathbf{X} alors construite n'est pas de plein rang $p + 1$ mais de rang p . La matrice $\mathbf{X}'\mathbf{X}$ n'est pas inversible et le modèle admet une infinité de solutions. Nous disons que les paramètres β_j ne sont pas *estimables* ou identifiables. En revanche, certaines fonctions (combinaisons linéaires) de ces paramètres sont estimables et appelées *contrastes*.

Dans le cas du modèle d'analyse de variance à *un* facteur, la solution la plus simple adoptée consiste à considérer un sous-ensemble des indicatrices ou de combinaisons des indicatrices engendrant le même sous-espace de façon à aboutir à une matrice inversible. Ceci conduit à considérer différents modèles associés à différentes *paramétrisation*. *Attention*, les paramètres β_j ainsi que la matrice \mathbf{X} prennent à chaque fois des significations différentes.

Un premier modèle (cell means model) s'écrit comme celui d'une régression linéaire multiple sans terme constant avec $\boldsymbol{\beta} = [\mu_1, \dots, \mu_J]'$ le vecteur des paramètres :

$$\begin{aligned}\mathbf{y} &= \beta_1 \mathbf{1}_1 + \dots + \beta_J \mathbf{1}_J + \boldsymbol{\varepsilon} \\ \mathbf{y} &= \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.\end{aligned}$$

Les calculs se présentent simplement mais les tests découlant de ce modèle conduiraient à étudier la nullité des paramètres alors que nous sommes intéressés par tester l'égalité des moyennes.

Une autre paramétrisation, considérant cette fois le vecteur $\beta = [\mu_J, \mu_1 - \mu_J, \dots, \mu_{J-1} - \mu_J]'$ conduit à écrire le modèle (base cell model) de régression avec terme constant :

$$\mathbf{y} = \beta_0 \mathbf{1} + \beta_1 \mathbf{1}_1 + \dots + \beta_{J-1} \mathbf{1}_{J-1} + \varepsilon.$$

C'est celle de SAS alors que d'autres logiciels considèrent des paramètres d'effet différentiel $\mu_j - \mu_{\cdot}$ par rapport à l'effet moyen $\mu_{\cdot} = 1/J \sum_{j=1}^J \mu_j$. Ce dernier est encore un modèle (group effect model) de régression linéaire avec terme constant mais dont les variables explicatives sont des différences d'indicateurs et avec $\beta = [\mu_{\cdot}, \mu_1 - \mu_{\cdot}, \dots, \mu_{J-1} - \mu_{\cdot}]'$:

$$\mathbf{y} = \beta_0 \mathbf{1} + \beta_1 (\mathbf{1}_1 - \mathbf{1}_J) + \dots + \beta_{J-1} (\mathbf{1}_{J-1} - \mathbf{1}_J) + \varepsilon.$$

7.3 Test

On désigne les différentes sommes des carrés des variations par :

$$\begin{aligned} \text{SST} &= \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\cdot\cdot})^2 = \sum_{j=1}^J \sum_{i=1}^{n_j} y_{ij}^2 - n \bar{y}_{\cdot\cdot}^2, \\ \text{SSW} &= \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_{\cdot j})^2 = \sum_{j=1}^J \sum_{i=1}^{n_j} y_{ij}^2 - \sum_{j=1}^J n_j \bar{y}_{\cdot j}^2, \\ \text{SSB} &= \sum_{j=1}^J n_j (\bar{y}_{\cdot j} - \bar{y}_{\cdot\cdot})^2 = \sum_{j=1}^J n_j \bar{y}_{\cdot j}^2 - n \bar{y}_{\cdot\cdot}^2, \end{aligned}$$

où "T" signifie totale, "W" (within) intra ou résiduelle, "B" (between) inter ou expliquée par la partition. Il est facile de vérifier que $\text{SST} = \text{SSB} + \text{SSW}$.

On considère alors l'hypothèse

$$H_0 : \mu_1 = \dots = \mu_J,$$

qui revient à dire que la moyenne est indépendante du niveau ou encore que le facteur n'a pas d'effet, contre l'hypothèse

$$H_1 : \exists(j, k) \text{ tel que } \mu_j \neq \mu_k$$

qui revient à reconnaître un effet ou une influence du facteur sur la variable Y .

Dans les modèles précédents, l'étude de cette hypothèse revient à comparer par un test de Fisher un modèle complet (les moyennes sont différentes) avec un modèle réduit supposant la nullité des paramètres β_j et donc l'égalité des moyennes à celle de la dernière cellule ou à la moyenne générale.

Les résultats nécessaires à la construction du test qui en découle sont résumés dans la table d'analyse de la variance :

<i>Source de variation</i>	<i>d.d.l.</i>	<i>Somme des carrés</i>	<i>Variance</i>	<i>F</i>
Modèle (inter)	$J - 1$	SSB	$MSB = SSB / (J - 1)$	MSB / MSW
Erreur (intra)	$n - J$	SSW	$MSW = SSW / (n - J)$	
Total	$n - 1$	SST		

Pratiquement, un programme de régression usuel permet de construire estimation et test de la nullité des β_j sauf pour le premier modèle qui doit tester l'égalité au lieu de la nullité des paramètres.

Dans le cas de deux classes ($J = 2$) on retrouve un test équivalent au test de Student de comparaison des moyennes de deux échantillons indépendants. Si l'hypothèse nulle est rejetée, la question suivante consiste à rechercher quelles sont les groupes ou cellules qui possèdent des moyennes significativement différentes. De nombreux tests et procédures ont été proposés dans la littérature pour répondre à cette question. Enfin, l'hypothèse importante du modèle induit par l'analyse de variance est l'homogénéité des variances de chaque groupe. Conjointement à l'estimation du modèle et en supposant la normalité, il peut être instructif de contrôler cette homogénéité par un test.

8 Analyse de covariance

L'*analyse de covariance* se situe encore dans le cadre général du modèle linéaire et où une variable quantitative est expliquée par plusieurs variables à la fois quantitatives et qualitatives. Dans les cas les plus complexes, on peut avoir plusieurs facteurs (variables qualitatives) avec une structure croisée ou hiérarchique ainsi que plusieurs variables quantitatives intervenant de manière linéaire ou polynômiale. Le principe général, dans un but explicatif ou décisionnel, est toujours d'estimer des modèles "*intra-groupes*" et de faire apparaître (tester) des effets différentiels "*inter-groupes*" des paramètres des régressions. Ainsi, dans le cas plus simple où seulement une variable parmi les explicatives est quantitative, nous sommes amenés à tester l'hétérogénéité des constantes et celle des pentes (interaction) entre différents modèles de régression linéaire.

Ce type de modèle permet donc, toujours avec un objectif prédictif, de s'intéresser à la modélisation d'une variable quantitative par un ensemble de variables explicatives à la fois quantitatives et qualitatives. La possible prise en compte d'interactions complique singulièrement la procédure de sélection de variables.

8.1 Modèle

Le modèle est explicité dans le cas élémentaire où une variable quantitative Y est expliquée par une variable qualitative T à J niveaux et une variable quantitative, appelée encore covariable, X . Pour chaque niveau j de T , on observe n_j valeurs $x_{1j}, \dots, x_{n_j j}$ de X et n_j valeurs $y_{1j}, \dots, y_{n_j j}$ de Y ; $n = \sum_{j=1}^J n_j$ est la taille de l'échantillon.

En pratique, avant de lancer une procédure de modélisation et tests, une démarche exploratoire s'appuyant sur une représentation en couleur (une par modalité j de T) du nuage de points croisant Y et X et associant les droites de régression permet de se faire une

idée sur les effets respectifs des variables : parallélisme des droites, étirement, imbrication des sous-nuages.

On suppose que les moyennes conditionnelles $E[Y|T]$, c'est-à-dire calculées à l'intérieur de chaque cellule, sont dans le sous-espace vectoriel engendré par les variables explicatives quantitatives, ici X . Ceci s'écrit :

$$y_{ij} = \beta_{0j} + \beta_{1j}x_{ij} + \varepsilon_{ij} ; \quad j = 1, \dots, J; \quad i = 1, \dots, n_j$$

où les ε_{ij} sont i.i.d. suivant une loi centrée de variance σ^2 qui sera supposée $\mathcal{N}(0, \sigma^2)$ pour la construction des tests.

Notons \mathbf{y} le vecteur des observations $[y_{ij}|i = 1, n_j; j = 1, J]'$ mis en colonne, \mathbf{x} le vecteur $[x_{ij}|i = 1, n_j; j = 1, J]'$, $\boldsymbol{\varepsilon} = [\varepsilon_{ij}|i = 1, n_j; j = 1, J]'$ le vecteur des erreurs, $\mathbf{1}_j$ les variables indicatrices des niveaux et $\mathbf{1}$ la colonne de 1s. On note encore $\mathbf{x}.\mathbf{1}_j$ le produit terme à terme des deux vecteurs, c'est-à-dire le vecteur contenant les observations de \mathbf{X} sur les individus prenant le niveau j de T et des zéros ailleurs.

La résolution simultanée des J modèles de régression est alors obtenue en considérant globalement le modèle :

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

dans lequel \mathbf{X} est la matrice $n \times 2J$ constituée des blocs $[\mathbf{1}_j|\mathbf{x}.\mathbf{1}_j]; j = 1, \dots, J$. L'estimation de ce modèle global conduit, par bloc, à estimer les modèles de régression dans chacune des cellules.

Comme pour l'analyse de variance, les logiciels opèrent une reparamétrisation faisant apparaître des effets différentiels par rapport au dernier niveau (SAS/GLM, SAS/INSIGHT) ou par rapport à un effet moyen (Systat), afin d'obtenir directement les bonnes hypothèses dans les tests. Ainsi, dans le premier cas, on considère la matrice de même rang (sans la J ème indicatrice)

$$\mathbf{X} = [\mathbf{1}|\mathbf{x}|\mathbf{1}_1|\dots|\mathbf{1}_{J-1}|\mathbf{x}.\mathbf{1}_1|\dots|\mathbf{x}.\mathbf{1}_{J-1}]$$

associée aux modèles :

$$y_{ij} = \beta_{0J} + (\beta_{0j} - \beta_{0J}) + \beta_{1J}x_{ij} + (\beta_{1j} - \beta_{1J})x_{ij} + \varepsilon_{ij} ; \quad j = 1, \dots, J-1; i = 1, \dots, n_j.$$

8.2 Tests

Différentes hypothèses sont alors testées en comparant le modèle complet

$$\begin{aligned} \mathbf{y} = & \beta_{0J}\mathbf{1} + (\beta_{01} - \beta_{0J})\mathbf{1}_1 + \dots + (\beta_{0J-1} - \beta_{0J})\mathbf{1}_{J-1} + \beta_{1J}\mathbf{x} + \\ & + (\beta_{11} - \beta_{1J})\mathbf{x}.\mathbf{1}_1 + \dots + (\beta_{1J-1} - \beta_{1J})\mathbf{x}.\mathbf{1}_{J-1} + \boldsymbol{\varepsilon} \end{aligned}$$

à chacun des modèles réduits :

- (i) $\mathbf{y} = \beta_{0J}\mathbf{1} + (\beta_{01} - \beta_{0J})\mathbf{1}_1 + \dots + (\beta_{0J-1} - \beta_{0J})\mathbf{1}_{J-1} + \beta_{1J}\mathbf{x} + \boldsymbol{\varepsilon}$
- (ii) $\mathbf{y} = \beta_{0J}\mathbf{1} + (\beta_{01} - \beta_{0J})\mathbf{1}_1 + \dots + (\beta_{0J-1} - \beta_{0J})\mathbf{1}_{J-1} +$
 $+ (\beta_{1j} - \beta_{1J})\mathbf{x}.\mathbf{1}_1 + \dots + (\beta_{1J-1} - \beta_{1J})\mathbf{x}.\mathbf{1}_{J-1} + \boldsymbol{\varepsilon}$
- (iii) $\mathbf{y} = \beta_{0J}\mathbf{1} + \beta_{1J}\mathbf{x} + (\beta_{1j} - \beta_{1J})\mathbf{x}.\mathbf{1}_1 + \dots + (\beta_{1J-1} - \beta_{1J})\mathbf{x}.\mathbf{1}_{J-1} + \boldsymbol{\varepsilon}$

par un test de Fisher. Ceci revient à considérer les hypothèses suivantes :

- H_0^i : pas d'interaction, $\beta_{11} = \dots = \beta_{1J}$, les droites partagent la même pente β_{1J} ,
- H_0^{ii} : $\beta_{1J}=0$,
- H_0^{iii} : $\beta_{01} = \dots = \beta_{0J}$, les droites partagent la même constante à l'origine β_{0J} .

On commence donc par évaluer i), si le test n'est pas significatif, on regarde ii) qui, s'il n'est pas non plus significatif, conduit à l'absence d'effet de la variable X . De même, toujours si i) n'est pas significatif, on s'intéresse à iii) pour juger de l'effet du facteur T .

8.3 Choix de modèle

Ce cadre théorique et les outils informatiques (SAS/GLM) permettent de considérer des modèles beaucoup plus complexes incluant plusieurs facteurs, plusieurs variables quantitatives, voire des polynômes de celles-ci, ainsi que les diverses interactions entre qualitatives et quantitatives. Le choix du “bon” modèle devient vite complexe d'autant que la stratégie dépend, comme pour la régression linéaire multiple, de l'objectif visé :

descriptif : des outils multidimensionnels descriptifs (ACP, AFD, AFCM...) s'avèrent souvent plus efficaces pour sélectionner, en première approche, un sous-ensemble de variables explicatives avant d'opérer une modélisation,

explicatif : de la prudence est requise d'autant que les hypothèses ne peuvent être évaluées de façon indépendante surtout si, en plus, des cellules sont déséquilibrées ou vides,

prédicatif : la recherche d'un modèle efficace, donc parcimonieux, peut conduire à négliger des interactions ou effets principaux lorsqu'une faible amélioration du R^2 le justifie et même si le test correspondant apparaît comme significatif. L'utilisation du C_p est théoriquement possible mais en général ce critère n'est pas calculé et d'utilisation délicate car nécessite la considération d'un “vrai” modèle de référence ou tout du moins d'un modèle de faible biais pour obtenir une estimation raisonnable de la variance de l'erreur. En revanche AIC et PRESS donnent des indications plus pertinentes. L'algorithme de recherche descendant est le plus couramment utilisé avec la contrainte suivante : *un effet principal n'est supprimé qu'à la condition qu'il n'apparaisse plus dans une interaction.*

8.4 Exemple

Les données, extraites de Jobson (1991), sont issues d'une étude marketing visant à étudier l'impact de différentes campagnes publicitaires sur les ventes de différents aliments. Un échantillon ou “panel” de familles a été constitué en tenant compte du lieu d'habitation ainsi que de la constitution de la famille. Chaque semaine, chacune de ces familles ont rempli un questionnaire décrivant les achats réalisés. Nous nous limitons ici à l'étude de l'impact sur la *consommation de lait* de quatre campagnes diffusées sur des chaînes locales de télévision. Quatre villes, une par campagne publicitaire, ont été choisies dans cinq différentes régions géographiques. Les consommations en lait par chacune des six familles par ville alors été mesurées (en dollars) après deux mois de campagne.

Les données se présentent sous la forme d'un tableau à 6 variables : la région géographique, les 4 consommations pour chacune des villes ou campagnes publicitaires diffusées, la taille de la famille. Cette situation est celle classique d'un modèle d'analyse de variance. Nous choisissons ici de conserver quantitative la variable taille de la famille et donc de modéliser

la consommation de lait par un modèle d'analyse de covariance plus *économique* en degrés de liberté moins de paramètres sont à estimer.

On s'intéresse à différents modèles de régression visant à expliquer la consommation en fonction de la taille de la famille conditionnellement au type de campagne publicitaire.

```
proc glm data=sasuser.milk;
class pub;
model consom=pub taille pub*taille;
run;
```

Les résultats ci-dessous conduiraient à conclure à une forte influence de la taille mais à l'absence d'influence du type de campagne. Les droites de régression ne semblent pas significativement différentes.

Source	DF	Type III SS	Mean Square	F Value	Pr > F
PUB	3	227.1807	75.7269	0.57	0.6377 (1)
TAILLE	1	40926.0157	40926.0157	306.57	0.0001 (2)
TAILLE*PUB	3	309.8451	103.2817	0.77	0.5111 (3)

-
- (1) Test de la significativité des différences des termes constants.
 (2) Test de l'influence du facteur quantitatif.
 (3) Test de la significativité des différences des pentes (interaction).
-

Néanmoins, pris d'un doute, le même calcul est effectué séparément pour chaque région :

```
proc glm data=sasuser.milk;
by region;
class pub;
model consom=pub taille pub*taille;
run;
```

Région	Source	DF	Type III SS	Mean Square	F Value	Pr > F
1	PUB	3	72.02974	24.00991	4.62	0.0164
	TAILLE	1	7178.32142	7178.32142	1380.25	0.0001
	TAILLE*PUB	3	217.37048	72.45683	13.93	0.0001
2	PUB	3	231.73422	77.24474	30.36	0.0001
	TAILLE	1	8655.25201	8655.25201	3402.34	0.0001
	TAILLE*PUB	3	50.15069	16.71690	6.57	0.0042
3	PUB	3	79.54688	26.51563	6.01	0.0061
	TAILLE	1	6993.30160	6993.30160	1585.35	0.0001
	TAILLE*PUB	3	173.19305	57.73102	13.09	0.0001
4	PUB	3	415.66664	138.55555	15.23	0.0001
	TAILLE	1	9743.37830	9743.37830	1071.32	0.0001
	TAILLE*PUB	3	361.39556	120.46519	13.25	0.0001
5	PUB	3	15.35494	5.11831	0.79	0.5168
	TAILLE	1	8513.28516	8513.28516	1314.71	0.0001
	TAILLE*PUB	3	52.75119	17.58373	2.72	0.0793

Il apparaît alors qu'à l'intérieur de chaque région (sauf région 5), les campagnes de publicité ont un effet tant sur la constante que sur la pente.

Ceci incite donc à se méfier des *interactions* (l'effet région compense l'effet publicité) et encourage à toujours conserver le facteur bloc (ici la région) dans une analyse de variance. Une approche complète, considérant *a priori* toutes les variables (3 facteurs), est ici nécessaire (cf. TP).

Compléter : choix automatique avec AIC.

9 Introduction

Dans ce chapitre, nous définissons le contexte pratique de la *régression logistique* qui s'intéressent plus particulièrement à la description ou l'explication d'observations constitués d'effectifs comme, par exemple, le nombre de succès d'une variable de Bernoulli lors d'une séquence d'essais. Nous laissons de côté le *modèle log-linéaire* (voir Agresti (1990) pour un exposé détaillé) qui vise à expliquer un nombre d'individus prenant une combinaison donnée de modalités de variables qualitatives ou niveaux de facteurs. Contrairement aux modèles du chapitre précédent basés sur l'hypothèse de normalité des observations, les lois concernées sont discrètes et associées à des dénombrements : loi de Poisson, binomiale, multinomiale. Néanmoins, tous ces modèles appartiennent à la famille du *modèle linéaire général* (annexe) et partagent à ce titre beaucoup d'aspects (estimation, tests, diagnostic) et dont la stratégie de mise en œuvre, similaire au cas gaussien, n'est pas reprise.

10 Odds et odds ratio

Une variable

Soit Y une variable qualitative à J modalités. On désigne la chance ou *l'odds*² de voir se réaliser la j ème modalité plutôt que la k ème par le rapport

$$\Omega_{jk} = \frac{\pi_j}{\pi_k}$$

où π_j est la probabilité d'apparition de la j ème modalité. Cette quantité est estimée par le rapport n_j/n_k des effectifs observés sur un échantillon. Lorsque la variable est binaire et suit une loi de Bernoulli de paramètre π , l'odds est le rapport $\pi/(1 - \pi)$ qui exprime une cote ou chance de gain.

Table de contingence

On considère maintenant une table de contingence 2×2 croisant deux variables qualitatives binaires X^1 et X^2 . les paramètres de la loi conjointe se mettent dans une matrice :

$$\begin{bmatrix} \pi_{11} & \pi_{12} \\ \pi_{21} & \pi_{22} \end{bmatrix}$$

où $\pi_{ij} = P[\{X^1 = i\} \text{ et } \{X^2 = j\}]$ est la probabilité d'occurrence de chaque combinaison.

- Dans la ligne 1, l'odds que la colonne 1 soit prise plutôt que la colonne 2 est :

$$\Omega_1 = \frac{\pi_{11}}{\pi_{12}}.$$

- Dans la ligne 2, l'odds que la colonne 1 soit prise plutôt que la colonne 2 est :

$$\Omega_2 = \frac{\pi_{21}}{\pi_{22}}.$$

²Il n'existe pas, même en Québécois, de traduction consensuelle de "odds".

On appelle odds ratio le rapport

$$\Theta = \frac{\Omega_1}{\Omega_2} = \frac{\pi_{11}\pi_{22}}{\pi_{12}\pi_{21}}.$$

Ce rapport prend la valeur 1 si les variables sont indépendantes, il est supérieur à 1 si les sujets de la ligne 1 ont plus de chances de prendre la première colonne que les sujets de la ligne 2 et inférieur à 1 sinon.

L'odds ratio est également défini pour deux lignes (a, b) et deux colonnes (c, d) quelconques d'une table de contingence croisant deux variables à J et K modalités. L'odds ratio est le rapport

$$\Theta_{abcd} = \frac{\Omega_a}{\Omega_b} = \frac{\pi_{ac}\pi_{bd}}{\pi_{ad}\pi_{bc}} \quad \text{estimé par l'odds ratio empirique} \quad \hat{\Theta}_{abcd} = \frac{n_{ac}n_{bd}}{n_{ad}n_{bc}}.$$

11 Régression logistique

11.1 Type de données

Cette section décrit la modélisation d'une variable qualitative Z à 2 modalités : 1 ou 0, succès ou échec, présence ou absence de maladie, panne d'un équipement, faillite d'une entreprise, bon ou mauvais client. . . . Les modèles de régression précédents adaptés à l'explication d'une variable quantitative ne s'appliquent plus directement car le régresseur linéaire usuel $\mathbf{X}\beta$ ne prend pas des valeurs simplement binaires. L'objectif est adapté à cette situation en cherchant à expliquer les probabilités

$$\pi = P(Z = 1) \quad \text{ou} \quad 1 - \pi = P(Z = 0),$$

ou plutôt une transformation de celles-ci, par l'observation conjointe des variables explicatives. L'idée est en effet de faire intervenir une fonction réelle monotone g opérant de $[0, 1]$ dans \mathbb{R} et donc de chercher un modèle linéaire de la forme :

$$g(\pi_i) = \mathbf{x}_i' \beta.$$

Il existe de nombreuses fonctions, dont le graphe présente une forme sigmoïdale et qui sont candidates pour remplir ce rôle, trois sont pratiquement disponibles dans les logiciels :

probit : g est alors la fonction inverse de la fonction de répartition d'une loi normale, mais son expression n'est pas explicite.

log-log avec g définie par

$$g(\pi) = \ln[-\ln(1 - \pi)]$$

mais cette fonction est dissymétrique.

logit est définie par

$$g(\pi) = \text{logit}(\pi) = \ln \frac{\pi}{1 - \pi} \quad \text{avec} \quad g^{-1}(x) = \frac{e^x}{1 + e^x}.$$

Plusieurs raisons, tant théoriques que pratiques, font préférer cette dernière solution. Le rapport $\pi/(1 - \pi)$, qui exprime une "cote", est l'*odds* et la *régression logistique* s'interprète donc comme la recherche d'une modélisation linéaire du "log odds" tandis que

les coefficients de certains modèles expriment des “odds ratio” c’est-à-dire l’influence d’un facteur qualitatif sur le risque (ou la chance) d’un échec (d’un succès) de Z .

Cette section se limite à la description de l’usage élémentaire de la régression logistique. Des compléments concernant l’explication d’une variable qualitative ordinaire (plusieurs modalités), l’intervention de variables explicatives avec effet aléatoire, l’utilisation de mesures répétées donc dépendantes, sont à rechercher dans la bibliographie.

11.2 Modèle binomial

On considère, pour $i = 1, \dots, I$, différentes valeurs *fixées* x_i^1, \dots, x_i^q des variables explicatives X^1, \dots, X^q . Ces dernières pouvant être des variables quantitatives ou encore des variables qualitatives, c’est-à-dire des facteurs issus d’une planification expérimentale.

Pour chaque groupe, c’est-à-dire pour chacune des combinaisons de valeurs ou facteurs, on réalise n_i observations ($n = \sum_{i=1}^I n_i$) de la variable Z qui se mettent sous la forme $y_1/n_1, \dots, y_I/n_I$ où y_i désigne le nombre de “succès” observés lors des n_i essais. On suppose que toutes les observations sont indépendantes et qu’à l’intérieur d’un même groupe, la probabilité π_i de succès est constante. Alors, la variable Y_i sachant n_i et d’espérance $E(Y_i) = n_i\pi_i$ suit une loi *binomiale* $\mathcal{B}(n_i, \pi_i)$ dont la fonction de densité s’écrit :

$$P(Y = y_i) = \binom{n_i}{y_i} \pi_i^{y_i} (1 - \pi_i)^{(n_i - y_i)}.$$

On suppose que le vecteur des fonctions *logit* des probabilités π_i appartient au sous-espace $\text{vect}\{X^1, \dots, X^q\}$ engendré par les variables explicatives :

$$\text{logit}(\pi_i) = \mathbf{x}_i' \boldsymbol{\beta} \quad i = 1, \dots, I$$

ce qui s’écrit encore

$$\pi_i = \frac{e^{\mathbf{x}_i' \boldsymbol{\beta}}}{1 + e^{\mathbf{x}_i' \boldsymbol{\beta}}} \quad i = 1, \dots, I.$$

Le vecteur des paramètres est estimé par maximisation de la log-vraisemblance. Il n’y a pas de solution analytique, celle-ci est obtenue par des méthodes numériques itératives (par exemple Newton Raphson) dont certaines reviennent à itérer des estimations de modèles de régression par moindres carrés généralisés avec des poids et des métriques adaptés à chaque itération.

L’optimisation fournit une estimation \mathbf{b} de $\boldsymbol{\beta}$, il est alors facile d’en déduire les estimations ou prévisions des probabilités π_i :

$$\hat{\pi}_i = \frac{e^{\mathbf{x}_i' \mathbf{b}}}{1 + e^{\mathbf{x}_i' \mathbf{b}}}$$

et ainsi celles des effectifs

$$\hat{y}_i = n_i \hat{\pi}_i.$$

Remarques

- i. La matrice \mathbf{X} issue de la planification expérimentale est construite avec les mêmes règles que celles utilisées dans le cadre de l'analyse de covariance mixant variables explicatives quantitatives et qualitatives. Ainsi, les logiciels gèrent avec plus ou moins de clarté le choix des variables indicatrices et donc des paramètres estimables ou contrastes associés.
- ii. La situation décrite précédemment correspond à l'observation de données *groupées*. Dans de nombreuses situations concrètes et souvent dès qu'il y a des variables explicatives quantitatives, les observations \mathbf{x}_i sont toutes distinctes. Ceci revient donc à fixer $n_i = 1; i = 1, \dots, I$ dans les expressions précédentes et la loi de Bernoulli remplace la loi binomiale. Certaines méthodes ne sont alors plus applicables et les comportements asymptotiques des distributions des statistiques de test ne sont plus valides, le nombre de paramètres tendant vers l'infini.

12 Choix de modèle

12.1 Recherche pas à pas

Principalement deux critères (test du rapport de vraisemblance et test de Wald, cf. bibliographie), sont utilisés de façon analogue au test de Fisher du modèle linéaire gaussien. Ils permettent de comparer un modèle avec un sous-modèle et d'évaluer l'intérêt de la présence des termes complémentaires. On suit ainsi une stratégie descendante à partir du modèle complet. L'idée est de supprimer, un terme à la fois, la composante d'interaction ou l'effet principal qui apparaît comme le moins significatif au sens du rapport de vraisemblance ou du test de Wald. Les tests présentent une structure hiérarchisée. SAS facilite cette recherche en produisant une décomposition (Type III) de ces indices permettant de comparer chacun des sous-modèles excluant un des termes avec le modèle les incluant tous.

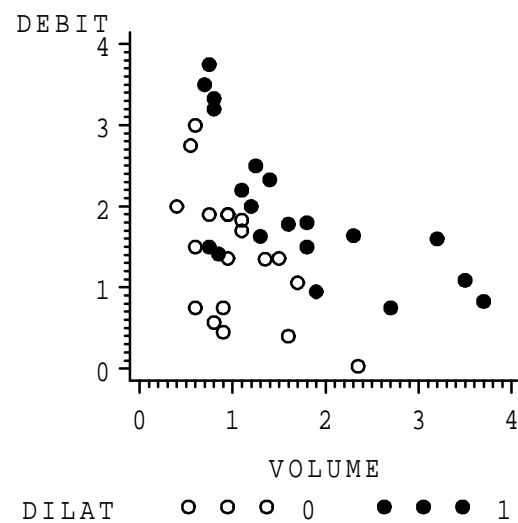
Attention, du fait de l'utilisation d'une transformation non linéaire (logit), même si des facteurs sont orthogonaux, aucune propriété d'orthogonalité ne peut être prise en compte pour l'étude des hypothèses. Ceci impose l'élimination des termes un par un et la ré-estimation du modèle. D'autre part, un terme principal ne peut être supprimé que s'il n'intervient plus dans des termes d'interaction.

Tout en déroulant l'algorithme de recherche ci-dessus, les logiciels calculent en plus l'AIC pour finaliser le choix pour une meilleure qualité prédictive.

13 Exemples

13.1 Débits×Volumes

On étudie l'influence du débit et du volume d'air inspiré sur l'occurrence (codée 1) de la dilatation des vaisseaux sanguins superficiels des membres inférieurs. Un graphique élémentaire représentant les modalités de Y dans les coordonnées de $X^1 \times X^2$ est toujours instructif. Il montre une séparation raisonnable et de bon augure des deux nuages de points. Dans le cas de nombreuses variables explicatives quantitatives, une analyse en composantes principales s'impose. Les formes des nuages représentés, ainsi que l'allure des distributions (étudiées préalablement), incitent dans ce cas à considérer par la suite les logarithmes des variables. Une variable **un** ne contenant que des "1" dénombrant le nombre d'essais est nécessaire dans la syntaxe de **genmod**. Les données sont en effet non groupées.

FIG. 2.4 – Nuage des modalités de Y dans les coordonnées des variables explicatives.

```
proc logistic data=sasuser.debvols;
model dilat=l_debit l_volume;
run;
proc genmod data=sasuser.debvols;
model dilat/un=l_debit l_volume/d=bin;
run;
```

The LOGISTIC Procedure

Criterion	Intercept and		Chi-Square for Covariates
	Only	Covariates	
AIC	56.040	35.216	.
SC	57.703	40.206	.
-2 LOG L	54.040	29.216(1)	24.824 with 2 DF (p=0.0001)
Score	.	.	16.635 with 2 DF (p=0.0002)

Variable	DF	Parameter(2)	Standard	Wald(3)	Pr >	Standardized	Odds
		Estimate	Error	Chi-Square	Chi-Square	Estimate	
INTERCPT	1	2.8782	1.3214	4.7443	0.0294	.	.
L_DEBIT	1	-4.5649	1.8384	6.1653	0.0130	-2.085068	0.010
L_VOLUME	1	-5.1796	1.8653	7.7105	0.0055	-1.535372	0.006

Cette procédure fournit des critères de choix de modèle dont la déviance (1), le vecteur \mathbf{b} des paramètres (2) et les statistiques des tests (3) comparant le modèle excluant un terme par rapport au modèle complet tel qu'il est décrit dans la commande.

The GENMOD Procedure

Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance	36	29.2156	0.8115 (1)
Scaled Deviance	36	29.2156	0.8115 (2)
Pearson Chi-Square	36	34.2516	0.9514 (3)
Scaled Pearson X2	36	34.2516	0.9514
Log Likelihood	.	-14.6078	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate (4)	Std Err	ChiSquare (5)	Pr>Chi
INTERCEPT	1	-2.8782	1.3214	4.7443	0.0294
L_DEBIT	1	4.5649	1.8384	6.1653	0.0130
L_VOLUME	1	5.1796	1.8653	7.7105	0.0055
SCALE (6)	0	1.0000	0.0000	.	.

-
- (1) Déviance du modèle par rapport au modèle saturé.
 (2) Déviance pondérée si le paramètre d'échelle est différent de 1 en cas de sur-dispersion.
 (3) Statistique de Pearson, voisine de la déviance, comparant le modèle au modèle saturé .
 (4) Paramètres du modèle.
 (5) Statistique des tests comparant le modèle excluant un terme par rapport au modèle complet.
 (6) Estimation du paramètre d'échelle si la quasi-vraisemblance est utilisée.
-

13.2 Données bancaires

Plusieurs stratégies peuvent être mises en œuvre sur les données bancaires. Les premières consistent à ne s'intéresser qu'aux variables quantitatives et à rechercher un "meilleur" modèle à l'aide de la procédure `logistic` en association avec l'un des trois algorithmes de sélection.

```
proc logistic data=sasuser.vispremt;
class (à compléter)
model carvpr =ager relat opgnbl--dnbjdl/selection=stepwise;
run;
```

Ainsi, l'algorithme qui procède par élimination retient finalement 14 des 20 variables. pour un taux de mal classés de 15,4%. Par sélection ou pas à pas, la même solution est proposée avec 12 variables pour un taux de 15,6%. *Attention*, ces taux, calculés sur l'échantillon ayant servi à estimer les paramètres, sont nécessairement trop optimistes.

À moins d'utiliser SAS Enterprise Miner, la prise en compte des variables qualitatives nécessitent une procédure de choix de modèle manuelle. Le module SAS/INSIGHT est alors le plus efficace pour réaliser une approche descendante avant de réestimer le modèle obtenu à l'aide de `genmod`.

```
proc genmod data=sasuser.vispremt ;
class sexec PRCS;
make 'OBSTATS' out=outglm;
model carvpr/poids = SEXEC PRCS OPGNBL MOYRVL BOPPNL
DNBJDL UEMNB XLGNB YLVNB JNTCA NPTAG / obstats d=bin;
run;
/* Estimation du taux de mal classés */
data prev ;
set outglm (keep=yvar1 pred);
if pred ge 0.5 then predy=1; else predy=0;
proc freq data=prev;
tables yvar1*predy/ nocol norow;
run;
```

Les résultats semblent alors bien meilleurs mais il faut rester prudent quant à la précision de cette estimation du taux d'erreur. On pourrait enfin s'interroger sur les qualités d'un modèle considérant toutes les variables qualitatives.

YVAR1 (CARVPR) PREDY

Frequency				
Percent	0	1	Total	
-----+-----+-----+				
0	659	53	712	
	61.65	4.96	66.60	
-----+-----+-----+				
1	70	287	357	
	6.55	26.85	33.40	
-----+-----+-----+				
Total	729	340	1069	
	68.19	31.81	100.00	

Chapitre 3

Erreur de prédiction

1 Introduction

La performance du modèle issu d'une méthode d'apprentissage s'évalue par sa *capacité de prédiction* dite encore de *généralisation*. La mesure de cette performance est très importante puisque, d'une part, elle permet d'opérer une *sélection de modèle* dans une famille associée à la méthode d'apprentissage utilisée et, d'autre part, elle guide le *choix de la méthode* en comparant chacun des modèles optimisés à l'étape précédente. Enfin, elle fournit, tous choix faits, une mesure de la qualité ou encore de la *confiance* que l'on peut accorder à la prévision en vue même, dans un cadre légal, d'une *certification*.

En dehors d'une situation expérimentale planifiée classique en Statistique, c'est-à-dire sans le secours de *modèles probabilistes*, c'est le cas, par principe, du *data mining*, trois types de stratégies sont proposés :

- un partage de l'échantillon (apprentissage, validation, test) afin de distinguer estimation du modèle et estimations de l'erreur de prédiction,
- une pénalisation de l'erreur d'ajustement par la complexité du modèle,
- un usage intensif du calcul (computational statistics) par la mise en œuvre de simulations.

. Le choix dépend de plusieurs facteurs dont la taille de l'échantillon initial, la complexité du modèle envisagé, la variance de l'erreur, la complexité des algorithmes c'est-à-dire le volume de calcul admissible. L'estimation de l'erreur de prédiction est donc un élément central de la mise en place de la stratégie du *data mining* telle qu'elle est décrite dans l'introduction (cf. chapitre 1 section).

2 Erreur de prédiction

2.1 Définition

Soit Y la variable à prédire, X la variable p -dimensionnelle ou l'ensemble des variables explicatives, F la loi conjointe de Y et X , $\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ un échantillon et

$$Y = \phi(X) + \varepsilon$$

le modèle à estimer avec $E(\varepsilon) = 0$, $\text{Var}(\varepsilon) = \sigma^2$ et ε indépendant de X ; X , comme chacun des \mathbf{x}_i , est de dimension p .

L'erreur de prévision est définie par

$$\mathcal{E}_P(\mathbf{z}, F) = E_F[Q(Y, \hat{\phi}(X))]$$

où Q est une *fonction perte*.

Si Y est quantitative, cette fonction perte est le plus généralement quadratique : $Q(y, \hat{y}) = (y - \hat{y})^2$, mais utilise parfois la valeur absolue : $Q(y, \hat{y}) = |y - \hat{y}|$. Cette dernière à l'avantage d'être plus *robuste*, car moins sensible aux valeurs extrêmes, mais nécessite des algorithmes d'optimisation plus complexes et pas nécessairement à solution unique.

Si Y est qualitative Q est une indicatrice de mal classé : $Q(y, \hat{y}) = \mathbf{1}_{\{y \neq \hat{y}\}}$.

Dans le cas quantitatif, l'estimation du modèle par minimisation de \mathcal{E}_P revient à une approximation de la fonction ϕ et la solution est l'*espérance conditionnelle* (connaissant l'échantillon) tandis que, dans le cas qualitatif, c'est la classe la plus probable désignée par le *mode conditionnel* qui est prédite.

2.2 Décomposition

L'erreur de prédiction se décompose dans le cas quantitatif¹. Considérons celle-ci en un point \mathbf{x}_0 .

$$\begin{aligned} \mathcal{E}_P(\mathbf{x}_0) &= E_F[(Y - \hat{\phi}(\mathbf{x}_0))^2 \mid X = \mathbf{x}_0] \\ &= \sigma^2 + [E_F \hat{\phi}(\mathbf{x}_0) - \phi(x)]^2 + E_F[\hat{\phi}(\mathbf{x}_0) - E_F \hat{\phi}(\mathbf{x}_0)]^2 \\ &= \sigma^2 + \text{Biais}^2 + \text{Variance}. \end{aligned}$$

Très généralement, plus un modèle (la famille des fonctions ϕ admissibles) est complexe, plus il est flexible et peu s'ajuster aux données observées et donc plus le biais est réduit. En revanche, la partie variance augmente avec le nombre de paramètres à estimer et donc avec cette complexité. L'enjeu, pour minimiser le risque quadratique ainsi défini, est donc de rechercher un meilleur compromis entre biais et variance : accepter de biaiser l'estimation comme par exemple en régression *ridge* pour réduire plus favorablement la variance.

2.3 Estimation

Le premier type d'estimation à considérer exprime la qualité d'ajustement du modèle sur l'échantillon observé. C'est justement, dans le cas quantitatif, ce critère qui est minimisé dans la recherche de moindres carrés. Ce ne peut être qu'une estimation biaisée, car trop *optimiste*, de l'erreur de prédiction ; elle est liée aux données qui ont servi à l'ajustement du modèle et est d'autant plus faible que le modèle est complexe. Cette estimation ne dépend que de la partie "biais" de l'erreur de prédiction et ne prend pas en compte la partie "variance" de la décomposition.

Cette estimation est notée :

$$\widehat{\mathcal{E}}_P = \frac{1}{n} \sum_{i=1}^n Q(y_i, \hat{\phi}(\mathbf{x}_i)).$$

¹Plusieurs décompositions concurrentes ont été proposées dans le cas qualitatif mais leur explicitation est moins claire.

C'est simplement le *taux de mal classés* dans le cas qualitatif. Des critères de risque plus sophistiqués sont envisagés dans un contexte bayésien si des probabilités *a priori* sont connues sur les classes ou encore des coûts de mauvais classement (cf. chapitre 4).

La façon la plus simple d'estimer sans biais l'erreur de prédiction consiste à calculer $\widehat{\mathcal{E}}_P$ sur un échantillon indépendant n'ayant pas participé à l'estimation du modèle. Ceci nécessite donc d'éclater l'échantillon en trois parties respectivement appelées *apprentissage*, *validation* et *test* :

$$\mathbf{Z} = \mathbf{Z}_{\text{Appr}} \cup \mathbf{Z}_{\text{Valid}} \cup \mathbf{Z}_{\text{Test}}.$$

- i. $\widehat{\mathcal{E}}_P(\mathbf{Z}_{\text{Appr}})$ est minimisée pour estimer un modèle,
- ii. $\widehat{\mathcal{E}}_P(\mathbf{Z}_{\text{Valid}})$ sert à la comparaison des modèles au sein d'une même famille afin de sélectionner celui qui minimise cette erreur,
- iii. $\widehat{\mathcal{E}}_P(\mathbf{Z}_{\text{Test}})$ est utilisée pour comparer entre eux les meilleurs modèles de chacune des méthodes considérées.

Cette solution n'est acceptable que si la taille de l'échantillon initiale est importante sinon :

- la qualité d'ajustement est dégradée car n est plus petit,
- la variance de l'estimation de l'erreur peut être importante et ne peut être estimée.

Si la taille de l'échantillon est insuffisante, le point ii ci-dessus : la sélection de modèle est basée sur un autre type d'estimation de l'erreur de prédiction faisant appel soit à une pénalisation soit à des simulations.

3 Estimation avec pénalisation

L'erreur de prédiction se décompose en :

$$\mathcal{E}_P = \widehat{\mathcal{E}}_P(\mathbf{Z}_{\text{Appr}}) + \text{Optim}$$

qui est l'estimation par resubstitution ou taux d'erreur apparent plus le biais par abus d'optimisme. Il s'agit donc d'estimer cette optimisme pour apporter une correction et ainsi une meilleure estimation de l'erreur recherchée. cette correction peut prendre plusieurs formes. Elle est liée à l'estimation de la variance dans la décomposition en biais et variance de l'erreur ou c'est encore une pénalisation associée à la complexité du modèle. Les estimateurs définis ci-dessous sont pour la plupart historiquement issus du modèle classique de régression multiple pour lequel il existe de nombreuses références mais ont été généralisés ou adaptés à d'autres méthodes en étendant la notion de nombre de degrés de libertés à des situations où le nombre de paramètres du modèle n'est pas explicite.

3.1 C_p , AIC, BIC

Le C_p de Mallows fut, historiquement, le premier critère visant à une meilleure estimation de l'erreur de prédiction que la seule considération de l'erreur d'ajustement (ou le R^2) dans le modèle linéaire. Son expression est détaillée dans le cas de la régression linéaire chapitre 2 sous l'hypothèse que le modèle complet à p variables est le "vrai" modèle. On montre (cf. Hastie et col. 2001), à des fins de comparaison qu'il peut aussi se mettre sous une forme équivalente :

$$C_p = \widehat{\mathcal{E}}_P + 2\frac{d}{n}s^2$$

où d est le nombre de paramètres du modèles, n le nombre d'observations, s^2 une estimation de la variance de l'erreur par un modèle de faible biais. Le critère d'information d'Akaike (AIC) se présente sous une forme similaire mais plus générale. Basé sur un critère de déviance, il s'applique en effet à tout modèle estimé par minimisation d'une log-vraisemblance \mathcal{L} . Ainsi, dans le cas de la régression logistique

$$\text{AIC} = -2\mathcal{L} + 2\frac{d}{n}.$$

Il suppose que la famille de densités considérées pour modéliser la loi de Y contient la "vraie" densité. Dans le cas gaussien en supposant la variance connue, moindres carrés et déviance coïncident, AIC est équivalent au C_p . Il est facile de choisir le modèle présentant le plus faible AIC parmi ceux considérés ce qui revient globalement à minimiser un critère de vraisemblance pénalisée. Celui-ci n'est vérifié qu'asymptotiquement d'où la motivation de proposer des critères modifiés (AICC) plus adaptés à de petits échantillons.

Pour les modèles non-linéaires ou plus complexes (non-paramétriques), le nombre q de paramètres doit être remplacé par une mesure de complexité $p(\alpha)$. Le critère se met alors sous une forme plus générale :

$$\text{AIC}(\alpha) = \widehat{\mathcal{E}}_P(\mathbf{x}_{\text{Appr}}) + 2\frac{p(\alpha)}{n}s^2.$$

Les modèles linéaires se mettent sous une forme : $\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}$, incluant les méthodes de régularisation (*ridge*) ou de lissage (spline) où la matrice \mathbf{H} dépend uniquement des \mathbf{x}_i . Dans ce cas, le nombre effectif de paramètres est défini comme la trace de la matrice \mathbf{H} : $d(\mathbf{H}) = \text{tr}(\mathbf{H})$. C'est encore q , le rang de \mathbf{X} c'est-à-dire le nombre vecteurs de base (le nombre de variables + 1) si \mathbf{H} est une matrice de projection orthogonale. Dans d'autres situations (perceptron), ce nombre de paramètres est plus difficile à contrôler car il fait intervenir les valeurs propres d'une matrice hessienne.

Une argumentation de type bayésienne conduit à un autre critère BIC (*Bayesian information criterion*) qui cherche, approximativement, le modèle associé à la plus grande probabilité *a posteriori* dans le cadre de la maximisation d'une log-vraisemblance. Il se met sous la forme :

$$\text{BIC} = \mathcal{L} + \log(n)d.$$

On montre, dans le cas gaussien et en supposant la variance connue que BIC est proportionnel à AIC avec le facteur 2 remplacé par $\log n$. Ainsi, dès que $n > e^2 \approx 7,4$, BIC tend à pénaliser plus lourdement les modèles complexes. Asymptotiquement, on montre que la probabilité pour BIC de choisir le bon modèle tend vers 1 lorsque n tend vers l'infini. Ce n'est pas le cas d'AIC qui tend alors à choisir des modèles trop complexes. Néanmoins à taille finie, BIC risque de se limiter à des modèles trop simples.

3.2 Dimension de Vapnik-Chernovenkis

(à compléter)

4 Estimation par simulation

4.1 Validation croisée

La validation croisée est conceptuellement simple, efficace et largement utilisée pour estimer une erreur moyennant un surplus de calcul. L'idée est d'itérer l'estimation de l'erreur sur plusieurs échantillons de *validation* puis d'en calculer la moyenne. C'est rapidement indispensable pour réduire la variance et améliorer la précision lorsque la taille de l'échantillon initial est trop réduite pour en extraire un échantillon de validation ou test de taille suffisante.

ALGORITHME 3.1 : Validation croisée

- Découper aléatoirement l'échantillon en K parts (K -fold) de tailles approximativement égales selon une loi uniforme ;
 - répéter K fois l'opération qui consiste à mettre de côté l'une des parties, estimer le modèle sur les $K - 1$ parties restantes, calculer l'erreur sur chacune des observations qui n'ont pas participé à l'estimation ;
 - moyenner toutes ces erreurs pour aboutir à l'estimation par validation croisée.
-

Plus précisément, soit $\tau : \{1, \dots, n\} \mapsto \{1, \dots, K\}$ la fonction d'indexation qui, pour chaque observation, donne l'attribution uniformément aléatoire de sa classe. L'estimation par *validation croisée* de l'erreur de prédiction est :

$$\widehat{\mathcal{E}_{CV}} = \frac{1}{n} \sum_{i=1}^n Q(y_i, \widehat{\phi}^{(-\tau(i))}(x_i))$$

où $\widehat{\phi}^{(-k)}$ désigne l'estimation de ϕ sans prendre en compte la k ème partie de l'échantillon.

Le choix $K = 10$ est le plus courant, c'est souvent celui par défaut des logiciels (Splus). Historiquement, la validation croisée a été introduite par Allen avec $K = n$ (*delete-one cross validation*). Ce dernier choix n'est possible que pour n relativement petit à cause du volume des calculs nécessaires et l'estimation de l'erreur présente une variance souvent importante car chacun des modèles estimés est trop similaire au modèle estimé avec toutes les observations. En revanche, si K est petit (*i.e.* $K = 5$), la variance sera plus faible mais le biais devient un problème dépendant de la façon dont la qualité de l'estimation se dégrade avec la taille de l'échantillon.

Minimiser l'erreur estimée par validation croisée est une approche largement utilisée pour optimiser le choix d'un modèle au sein d'une famille paramétrée. $\widehat{\phi}$ est défini par $\widehat{\theta} = \arg \min_{\theta} \widehat{E}_{CV}(\theta)$.

4.2 Bootstrap

Cette section plus technique décrit des outils encore peu présents dans les logiciels commerciaux, elle peut être sautée en première lecture.

Introduction

L'idée, d'approcher par simulation (*Monte Carlo*) la distribution d'un estimateur lorsque l'on ne connaît pas la loi de l'échantillon ou, plus souvent, lorsque l'on ne peut pas supposer

qu'elle est gaussienne, est l'objectif même du *bootstrap* (Efron, 1982).

Le principe fondamental de cette technique de rééchantillonnage est de substituer, à la distribution de probabilité inconnue F , dont est issu l'échantillon d'apprentissage, la distribution empirique F_n qui donne un poids $1/n$ à chaque réalisation. Ainsi on obtient un échantillon de taille n dit *échantillon bootstrap* selon la distribution empirique F_n par n tirages aléatoires avec remise parmi les n observations initiales.

Il est facile de construire un grand nombre d'échantillons bootstrap (*i.e.* $B = 100$) sur lesquels calculer l'estimateur concerné. La loi simulée de cet estimateur est une approximation asymptotiquement convergente sous des hypothèses raisonnables² de la loi de l'estimateur. Cette approximation fournit ainsi des estimations du biais, de la variance, donc d'un risque quadratique, et même des intervalles de confiance (avec B beaucoup plus grand) de l'estimateur sans hypothèse (normalité) sur la vraie loi. Les grands principes de cette approche sont rappelés en annexe B.

Estimateur naïf

Soit \mathbf{z}^* un échantillon bootstrap des données :

$$\mathbf{z}^* = \{(\mathbf{x}_1^*, y_1^*), \dots, (\mathbf{x}_n^*, y_n^*)\}.$$

L'estimateur *plug-in* de l'erreur de prédiction $\mathcal{E}_P(\mathbf{z}, F)$, pour lequel la distribution F est remplacée par la distribution empirique \hat{F} (cf. section B1.1) est défini par :

$$\mathcal{E}_P(\mathbf{z}^*, \hat{F}) = \frac{1}{n} \sum_{i=1}^n nQ(y_i, \phi_{\mathbf{z}^*}(x_i))$$

où $\phi_{\mathbf{z}^*}$ désigne l'estimation de ϕ à partir de l'échantillon bootstrap. Il conduit à l'estimation bootstrap de l'erreur moyenne de prédiction $E_F[\mathcal{E}_P(\mathbf{z}, F)]$ par

$$\mathcal{E}_{\text{Boot}} = E_{\hat{F}}[\mathcal{E}_P(\mathbf{z}^*, \hat{F})] = E_{\hat{F}} \left[\frac{1}{n} \sum_{i=1}^n nQ(y_i, \phi_{\mathbf{z}^*}(x_i)) \right].$$

Cette estimation est approchée par simulation :

$$\widehat{\mathcal{E}_{\text{Boot}}} = \frac{1}{B} \sum_{b=1}^B \frac{1}{n} \sum_{i=1}^n nQ(y_i, \phi_{\mathbf{z}^{*b}}(x_i)).$$

L'estimation ainsi construite de l'erreur de prédiction est généralement biaisée par optimisme car, au gré des simulations, les mêmes observations (\mathbf{x}_i, y_i) apparaissent à la fois dans l'estimation du modèle et dans celle de l'erreur. D'autres approches visent à corriger ce biais.

Estimateur *out-of-bag*

La première s'inspire simplement de la validation croisée. Elle considère d'une part les observations tirées dans l'échantillon bootstrap et, d'autre part, celles qui sont laissées de

²Échantillon indépendant de même loi et estimateur indépendant de l'ordre des observations.

côté pour l'estimation du modèle mais retenue pour l'estimation de l'erreur.

$$\widehat{\mathcal{E}}_{\text{oob}} = \frac{1}{n} \sum_{i=1}^n \frac{1}{B_i} \sum_{b \in K_i} Q(y_i, \phi_{\mathbf{z}^{*b}}(x_i))$$

où K_i est l'ensemble des indices b des échantillons bootstrap ne contenant pas la i ème observation à l'issue des B simulations et $B_i = |K_i|$ le nombre de ces échantillons ; B doit être suffisamment grand pour que toute observation n'ait pas été tirée au moins une fois ou bien les termes avec $K_i = 0$ sont supprimés.

L'estimation $\widehat{\mathcal{E}}_{\text{oob}}$ résout le problème d'un biais optimiste auquel est confrontée $\widehat{\mathcal{E}}_{\text{Boot}}$ mais n'échappe pas au biais introduit par la réduction tel qu'il est signalé pour l'estimation par validation croisée $\widehat{\mathcal{E}}_{\text{CV}}$. C'est ce qui a conduit Efron et Tibshirani (1997) à proposer des correctifs.

Estimateur .632-bootstrap

La probabilité qu'une observation soit tirée dans un échantillon bootstrap est

$$P[\mathbf{x}_i \in \mathbf{x}^{*b}] = 1 - \left(1 - \frac{1}{n}\right)^n \approx 1 - \frac{1}{e} \approx 0,632.$$

Très approximativement, la dégradation de l'estimation provoquée par le bootstrap et donc la surévaluation de l'erreur sont analogues à celle de la validation croisée avec $K = 2$. À la suite d'un raisonnement trop long pour être reproduit ici, Efron et Tibshirani (1997) proposent de compenser excès d'optimisme du taux apparent d'erreur et excès de pessimisme du bootstrap *out-of-bag* par une combinaison :

$$\widehat{\mathcal{E}}_{.632} = 0,368 \times \widehat{\mathcal{E}}_P + 0,632 \times \widehat{\mathcal{E}}_{\text{oob}}.$$

4.3 Remarques

- Toutes les estimations de l'erreur de prédiction considérées (pénalisation, validation croisée, bootstrap) sont asymptotiquement équivalentes et il n'est pas possible de savoir laquelle concrètement sera, à n fini, la plus précise. Une large part d'arbitraire ou d'"expérience" préside donc le choix d'une estimation plutôt qu'une autre.
- Conceptuellement, le bootstrap est plus compliqué et pratiquement encore peu utilisé. Néanmoins, cet outil joue un rôle central dans les algorithmes récents de combinaison de modèles (cf. chapitre 7) en association avec une estimation *out-of-bag* de l'erreur. Il ne peut être négligé.
- L'estimateur .632-bootstrap pose des problèmes en situation de sur-ajustement aussi les mêmes auteurs ont proposé un rectificatif complémentaire noté *.632+bootstrap*.

Ce qu'il faut retenir en conclusion, c'est que l'estimation d'une erreur de prédiction est une opération délicate aux conséquences importantes. Il est donc nécessaire

- d'utiliser le *même estimateur* pour comparer l'efficacité de deux méthodes,
- de se montrer très prudent, en dehors de tout système d'hypothèses probabilistes, sur le caractère absolu d'une estimation dans l'objectif d'une certification.

Dans ces deux dernières situations, le recours à un échantillon test de bonne taille est difficilement contournable alors qu'en situation de choix de modèle au sein d'une même famille, un estimateur (petit échantillon de validation, validation croisée) plus économique

est adapté en supposant implicitement que le biais induit est identique d'un modèle à l'autre.

Chapitre 4

Analyse Discriminante Décisionnelle

1 Introduction

L'objet de ce chapitre est l'explication d'une variable qualitative Y à m modalités par p variables quantitatives $X^j, j = 1, \dots, p$ observées sur un même échantillon Ω de taille n . L'objectif de l'analyse discriminante décisionnelle dépasse le simple cadre descriptif de l'analyse factorielle discriminante (AFD). Disposant d'un nouvel individu (ou de plusieurs, c'est la même chose) sur lequel on a observé les X^j mais pas Y , il s'agit maintenant de *décider* de la modalité \mathcal{T}_ℓ de Y (ou de la classe correspondante) de ce nouvel individu. On parle aussi de problème d'*affectation*. L'ADD s'applique donc également à la situation précédente de la régression logistique ($m = 2$) mais aussi lorsque le nombre de classes est plus grand que 2.

Pour cela, on va définir et étudier dans ce chapitre des *règles de décision* (ou d'affectation) et donner ensuite les moyens de les évaluer sur un seul individu ; $\mathbf{x} = (x^1, \dots, x^p)$ désigne les observations des variables explicatives sur cet individu, $\{\mathbf{g}_\ell; \ell = 1, \dots, m\}$ les barycentres des classes calculés sur l'échantillon et $\bar{\mathbf{x}}$ le barycentre global.

La matrice de covariance empirique se décompose en

$$\mathbf{S} = \mathbf{S}_e + \mathbf{S}_r.$$

où \mathbf{S}_r est appelée variance intraclasse (within) ou résiduelle :

$$\mathbf{S}_r = \overline{\mathbf{X}}_r' \mathbf{D} \overline{\mathbf{X}}_r = \sum_{\ell=1}^m \sum_{i \in \Omega_\ell} w_i (\mathbf{x}_i - \mathbf{g}_\ell)(\mathbf{x}_i - \mathbf{g}_\ell)',$$

et \mathbf{S}_e la variance interclasse (between) ou expliquée :

$$\mathbf{S}_e = \overline{\mathbf{G}}' \mathbf{D} \overline{\mathbf{G}} = \overline{\mathbf{X}}_e' \mathbf{D} \overline{\mathbf{X}}_e = \sum_{\ell=1}^m \overline{w}_\ell (\mathbf{g}_\ell - \bar{\mathbf{x}})(\mathbf{g}_\ell - \bar{\mathbf{x}})'.$$

2 Règle de décision issue de l'AFD

2.1 Cas général : m quelconque

DÉFINITION 4.1. — On affectera l'individu x à la modalité de Y minimisant :

$$d_{\mathbf{S}_r^{-1}}^2(\mathbf{x}, \mathbf{g}_\ell), \ell = 1, \dots, m.$$

Cette distance se décompose en

$$d_{\mathbf{S}_r^{-1}}^2(\mathbf{x}, \mathbf{g}_\ell) = \|\mathbf{x} - \mathbf{g}_\ell\|_{\mathbf{S}_r^{-1}}^2 = (\mathbf{x} - \mathbf{g}_\ell)' \mathbf{S}_r^{-1} (\mathbf{x} - \mathbf{g}_\ell)$$

et le problème revient donc à maximiser

$$\mathbf{g}_\ell' \mathbf{S}_r^{-1} \mathbf{x} - \frac{1}{2} \mathbf{g}_\ell' \mathbf{S}_r^{-1} \mathbf{g}_\ell.$$

Il s'agit bien d'une règle linéaire en \mathbf{x} car elle peut s'écrire : $\mathbf{A}_\ell \mathbf{x} + \mathbf{b}_\ell$.

2.2 Cas particulier : $m = 2$

Dans ce cas, la dimension r de l'AFD vaut 1. Il n'y a qu'une seule valeur propre non nulle λ_1 , un seul vecteur discriminant v^1 et un seul axe discriminant Δ_1 . Les 2 barycentres \mathbf{g}_1 et \mathbf{g}_2 sont sur Δ_1 , de sorte que v^1 est colinéaire à $\mathbf{g}_1 - \mathbf{g}_2$.

L'application de la règle de décision permet d'affecter \mathbf{x} à \mathcal{T}_1 si :

$$\mathbf{g}_1' \mathbf{S}_r^{-1} \mathbf{x} - \frac{1}{2} \mathbf{g}_1' \mathbf{S}_r^{-1} \mathbf{g}_1 > \mathbf{g}_2' \mathbf{S}_r^{-1} \mathbf{x} - \frac{1}{2} \mathbf{g}_2' \mathbf{S}_r^{-1} \mathbf{g}_2$$

c'est-à-dire encore si

$$(\mathbf{g}_1 - \mathbf{g}_2)' \mathbf{S}_r^{-1} \mathbf{x} > (\mathbf{g}_1 - \mathbf{g}_2)' \mathbf{S}_r^{-1} \frac{\mathbf{g}_1 + \mathbf{g}_2}{2}.$$

Remarque

La règle de décision liée à l'AFD est simple mais elle est limitée et insuffisante notamment si les variances des classes ne sont pas identiques. De plus, elle ne tient pas compte de l'échantillonnage pour \mathbf{x} : tous les groupes n'ont pas nécessairement la même probabilité d'occurrence.

3 Règle de décision bayésienne

3.1 Introduction

Dans cette optique, on considère que la variable Y , qui indique le groupe d'appartenance d'un individu, prend ses valeurs dans $\{\mathcal{T}_1, \dots, \mathcal{T}_m\}$ et est munie d'une loi de probabilité π_1, \dots, π_m . Les probabilités $\pi_\ell = P[\mathcal{T}_\ell]$ représentent les probabilités *a priori* des classes ou groupes ω_ℓ . On suppose que les vecteurs \mathbf{x} des observations des variables explicatives suivent, connaissant leur classe, une loi de densité

$$f_\ell(\mathbf{x}) = P[\mathbf{x} \mid \mathcal{T}_\ell]$$

par rapport à une mesure de référence¹.

3.2 Définition

Une règle de décision est une application δ de Ω dans $\{\mathcal{T}_1, \dots, \mathcal{T}_m\}$ qui, à tout individu, lui affecte une classe connaissant \mathbf{x} . Sa définition dépend du contexte de l'étude et prend en compte la

- connaissance ou non de coûts de mauvais classement,
- connaissance ou non des lois *a priori* sur les classes,
- nature aléatoire ou non de l'échantillon.

On désigne par $c_{\ell | k}$ le coût du classement dans \mathcal{T}_ℓ d'un individu de \mathcal{T}_k . Le *risque de Bayes* d'une règle de décision δ exprime alors le coût moyen :

$$R_\delta = \sum_{k=1}^m \pi_k \sum_{\ell=1}^m c_{\ell | k} \int_{\{\mathbf{x} | \delta(\mathbf{x})=\mathcal{T}_\ell\}} f_k(\mathbf{x}) d\mathbf{x}$$

où $\int_{\{\mathbf{x} | \delta(\mathbf{x})=\mathcal{T}_\ell\}} f_k(\mathbf{x}) d\mathbf{x}$ représente la probabilité d'affecté \mathbf{x} à \mathcal{T}_ℓ alors qu'il est dans \mathcal{T}_k .

3.3 Coûts inconnus

L'estimation des coûts n'est pas du ressort de la Statistique et, s'ils ne sont pas connus, on suppose simplement qu'ils sont tous égaux. La minimisation du risque ou règle de Bayes revient alors à affecter tout \mathbf{x} à la classe la plus probable c'est-à-dire à celle qui maximise la probabilité conditionnelle *a posteriori* : $P[\mathcal{T}_\ell | \mathbf{x}]$. Par le théorème de Bayes, on a :

$$P[\mathcal{T}_\ell | \mathbf{x}] = \frac{P[\mathcal{T}_\ell \text{ et } \mathbf{x}]}{P[\mathbf{x}]} = \frac{P[\mathcal{T}_\ell] \cdot P[\mathbf{x} | \mathcal{T}_\ell]}{P[\mathbf{x}]}$$

avec le principe des probabilités totales : $P[\mathbf{x}] = \sum_{\ell=1}^m P[\mathcal{T}_\ell] \cdot P[\mathbf{x} | \mathcal{T}_\ell]$.

Comme $P[\mathbf{x}]$ ne dépend pas de ℓ , la règle consistera à choisir \mathcal{T}_ℓ maximisant

$$P[\mathcal{T}_\ell] \cdot P[\mathbf{x} | \mathcal{T}_\ell] = \pi_\ell \cdot P[\mathbf{x} | \mathcal{T}_\ell];$$

$P[\mathbf{x} | \mathcal{T}_\ell]$ est la probabilité d'observer \mathbf{x} au sein de la classe \mathcal{T}_ℓ . Pour une loi discrète, il s'agit d'une probabilité du type $P[\mathbf{x} = \mathbf{x}_k^l | \mathcal{T}_\ell]$ et d'une densité $f(\mathbf{x} | \mathcal{T}_\ell)$ pour une loi continue. Dans tous les cas nous utiliserons la notation $f_\ell(\mathbf{x})$.

La règle de décision s'écrit finalement sous la forme :

$$\delta(\mathbf{x}) = \arg \max_{\ell=1, \dots, m} \pi_\ell f_\ell(\mathbf{x}).$$

3.4 Détermination des *a priori*

Les probabilités *a priori* π_ℓ peuvent effectivement être connues *a priori* : proportions de divers groupes dans une population, de diverses maladies... ; sinon elles sont estimées sur l'échantillon d'apprentissage :

$$\hat{\pi}_\ell = w_\ell = \frac{n_\ell}{n} \quad (\text{si tous les individus ont le même poids})$$

à condition qu'il soit bien un échantillon aléatoire susceptible de fournir des estimations correctes des fréquences. Dans le cas contraire il reste à considérer tous les π_ℓ égaux.

¹La mesure de Lebesgues pour des variables réelles, celle de comptage pour des variables qualitatives

3.5 Cas particuliers

- Dans le cas où les probabilités *a priori* sont égales, c'est par exemple le cas du choix de probabilités non informatives, la règle de décision bayésienne revient alors à maximiser $f_\ell(\mathbf{x})$ qui est la vraisemblance, au sein de \mathcal{T}_ℓ , de l'observation \mathbf{x} . La règle consiste alors à choisir la classe pour laquelle cette vraisemblance est maximum.
- Dans le cas où $m = 2$, on affecte \mathbf{x} à \mathcal{T}_1 si :

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{\pi_2}{\pi_1}$$

faisant ainsi apparaître un rapport de vraisemblance. D'autre part, l'introduction de coûts de mauvais classement différents selon les classes amène à modifier la valeur limite π_2/π_1 .

Finalement, il reste à estimer les densités conditionnelles $f_\ell(\mathbf{x})$. Les différentes méthodes d'estimation considérées conduisent aux méthodes classiques de discrimination bayésienne objets des sections suivantes.

4 Règle bayésienne avec modèle normal

On suppose dans cette section que, conditionnellement à \mathcal{T}_ℓ , $\mathbf{x} = (x_1, \dots, x_p)$ est l'observation d'un vecteur aléatoire gaussien $\mathcal{N}(\boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}_\ell)$; $\boldsymbol{\mu}_\ell$ est un vecteur de \mathbb{R}^p et $\boldsymbol{\Sigma}_\ell$ une matrice $(p \times p)$ symétrique et définie-positive. La densité de la loi, au sein de la classe \mathcal{T}_ℓ , s'écrit donc :

$$f_\ell(\mathbf{x}) = \frac{1}{\sqrt{2\pi}(\det(\boldsymbol{\Sigma}_\ell))^{1/2}} \exp \left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_\ell)' \boldsymbol{\Sigma}_\ell^{-1} (\mathbf{x} - \boldsymbol{\mu}_\ell) \right].$$

L'affectation de \mathbf{x} à une classe se fait en maximisant $\pi_\ell f_\ell(\mathbf{x})$ par rapport à l soit encore la quantité :

$$\ln(\pi_\ell) - \frac{1}{2} \ln(\det(\boldsymbol{\Sigma}_\ell)) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_\ell)' \boldsymbol{\Sigma}_\ell^{-1} (\mathbf{x} - \boldsymbol{\mu}_\ell).$$

4.1 Hétéroscédasticité

Dans le cas général, il n'y a pas d'hypothèse supplémentaire sur la loi de \mathbf{x} et donc les matrices $\boldsymbol{\Sigma}_\ell$ sont fonction de ℓ . Le critère d'affectation est alors *quadratique* en \mathbf{x} . Les probabilités π_ℓ sont supposées connues mais il est nécessaire d'estimer les moyennes $\boldsymbol{\mu}_\ell$ ainsi que les covariances $\boldsymbol{\Sigma}_\ell$ en maximisant, compte tenu de l'hypothèse de normalité, la vraisemblance. Ceci conduit à estimer la moyenne

$$\widehat{\boldsymbol{\mu}}_\ell = \mathbf{g}_\ell$$

par la moyenne empirique de \mathbf{x} dans la classe l pour l'échantillon d'apprentissage et $\boldsymbol{\Sigma}_\ell$ par la matrice de covariance empirique \mathbf{S}_{Rl}^* :

$$\mathbf{S}_{Rl}^* = \frac{1}{n_\ell - 1} \sum_{i \in \Omega_\ell} (\mathbf{x}_i - \mathbf{g}_\ell)(\mathbf{x}_i - \mathbf{g}_\ell)'$$

pour ce même échantillon.

4.2 Homoscédasticité

On suppose dans ce cas que les lois de chaque classe partagent la même structure de covariance $\Sigma_\ell = \Sigma$. Supprimant les termes indépendants de l , le critère à maximiser devient

$$\ln(\pi_\ell) - \frac{1}{2} \boldsymbol{\mu}'_\ell \Sigma_\ell^{-1} \boldsymbol{\mu}_\ell + \boldsymbol{\mu}'_\ell \Sigma_\ell^{-1} \mathbf{x}$$

qui est cette fois *linéaire* en \mathbf{x} . Les moyennes $\boldsymbol{\mu}_\ell$ sont estimées comme précédemment tandis que Σ est estimée par la matrice de covariance intra empirique :

$$\mathbf{S}_R^* = \frac{1}{n-m} \sum_{\ell=1}^m \sum_{i \in \Omega_\ell} (\mathbf{x}_i - \mathbf{g}_\ell)(\mathbf{x}_i - \mathbf{g}_\ell)'.$$

Si, de plus, les probabilités π_ℓ sont égales, après estimation le critère s'écrit :

$$\bar{\mathbf{x}}'_\ell \mathbf{S}_R^{*-1} \mathbf{x} - \frac{1}{2} \bar{\mathbf{x}}'_\ell \mathbf{S}_R^{*-1} \bar{\mathbf{x}}_\ell.$$

On retrouve alors le critère de la section 2 issu de l'AFD.

4.3 Commentaire

Les hypothèses : normalité, éventuellement l'homoscédasticité, doivent être vérifiées par la connaissance *a priori* du phénomène ou par une étude préalable de l'échantillon d'apprentissage. L'hypothèse d'homoscédasticité, lorsqu'elle est vérifiée, permet de réduire très sensiblement le nombre de paramètres à estimer et d'aboutir à des estimateurs plus fiables car de variance moins élevée. Dans le cas contraire, l'échantillon d'apprentissage doit être de taille importante.

5 Règle bayésienne avec estimation non paramétrique

5.1 Introduction

En Statistique, on parle d'estimation non paramétrique ou fonctionnelle lorsque le nombre de paramètres à estimer est infini. L'objet statistique à estimer est alors une fonction par exemple de régression $y = f(x)$ ou encore une densité de probabilité. Dans ce cas, au lieu de supposer qu'on a affaire à une densité de type connu (normale) dont on estime les paramètres, on cherche une estimation \hat{f} de la fonction de densité f . Pour tout x de \mathbb{R} , $f(x)$ est donc estimée par $\hat{f}(x)$.

Cette approche très souple a l'avantage de ne pas nécessiter d'hypothèse particulière sur la loi (seulement la régularité de f pour de bonnes propriétés de convergence), en revanche elle n'est applicable qu'avec des échantillons de grande taille d'autant plus que le nombre de dimensions p est grand (*curse of dimensionality*).

Dans le cadre de l'analyse discriminante, ces méthodes permettent d'estimer directement les densités $f_\ell(\mathbf{x})$. On considère ici deux approches : la méthode du noyau et celle des k plus proches voisins.

5.2 Méthode du noyau

Estimation de densité

Soit y_1, \dots, y_n n observations équipondérées d'une v.a.r. continue Y de densité f inconnue. Soit $K(y)$ (le *noyau*) une densité de probabilité unidimensionnelle (sans rapport avec f) et h un réel strictement positif. On appelle estimation de f par la méthode du noyau la fonction

$$\hat{f}(y) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{y - y_i}{h}\right).$$

Il est immédiat de vérifier que

$$\forall y \in \mathbb{R}, \hat{f}(y) \geq 0 \quad \text{et} \quad \int_{-\infty}^{+\infty} \hat{f}(y) dy = 1;$$

h est appelé *largeur de fenêtre* ou paramètre de *lissage*; plus h est grand, plus l'estimation \hat{f} de f est régulière. Le noyau K est choisi centré en 0, unimodal et symétrique. Les cas les plus usuels sont la densité gaussienne, celle uniforme sur $[-1, 1]$ ou triangulaire : $K(x) = [1 - |x|]\mathbf{1}_{[-1, 1]}(x)$. La forme du noyau n'est pas très déterminante sur la qualité de l'estimation contrairement à la valeur de h .

Application à l'analyse discriminante

La méthode du noyau est utilisée pour calculer une estimation non paramétrique de chaque densité $f_\ell(\mathbf{x})$ qui sont alors des fonctions définies dans \mathbb{R}^p . Le noyau K^* dont donc être choisi multidimensionnel et

$$\hat{f}_\ell(\mathbf{x}) = \frac{1}{n_\ell h^p} \sum_{i \in \Omega_\ell} K^*\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right).$$

Un noyau multidimensionnel peut être défini à partir de la densité usuelle de lois : multinormale $\mathcal{N}_p(0, \Sigma_p)$ ou uniforme sur la sphère unité ou encore par produit de noyaux unidimensionnels :

$$K^*(\mathbf{x}) = \prod_{j=1}^p K(x^j).$$

5.3 k plus proches voisins

Cette méthode d'affectation d'un vecteur \mathbf{x} consiste à enchaîner les étapes décrites dans l'algorithme ci-dessous.

ALGORITHME 4.1 : **k-nn**

- Choisir d'un entier $k : 1 \leq k \leq n$.
- Calculer les distances $d_{\mathbf{M}}(\mathbf{x}, \mathbf{x}_i)$, $i = 1, \dots, n$ où \mathbf{M} est la métrique de Mahalanobis c'est-à-dire la matrice inverse de la matrice de variance (ou de variance intra).
- Retenir les k observations $\mathbf{x}_{(1)}, \dots, \mathbf{x}_{(k)}$ pour lesquelles ces distances sont les plus petites.

- Compter les nombres de fois k_1, \dots, k_m que ces k observations apparaissent dans chacune des classes.
- Estimer les densités par

$$\hat{f}_\ell(\mathbf{x}) = \frac{k_\ell}{kV_k(\mathbf{x})};$$

où $V_k(\mathbf{x})$ est le volume de l'ellipsoïde $\{\mathbf{z} | (\mathbf{z} - \mathbf{x})' \mathbf{M} (\mathbf{z} - \mathbf{x}) = d_{\mathbf{M}}(\mathbf{x}, \mathbf{x}_{(k)})\}$.

Pour $k = 1$, \mathbf{x} est affecté à la classe du plus proche élément.

Comme toute technique, celles présentées ci-dessus nécessitent le réglage d'un paramètre (largeur de fenêtre, nombre de voisins considérés). Ce choix s'apparente à un choix de modèle et nécessite le même type d'approche à savoir l'optimisation d'un critère (erreur de classement, validation croisée (cf. chapitre 3)).

6 Exemple

Une première étape de traitement des données bancaires permet tout d'abord de sélectionner par élimination un sous-ensemble des variables à l'aide de la procédure `stepdisc`. La variable qualitative `sexe` est considérée comme une variable quantitative (0, 1). Ceci pourrait, abusif mais fréquent en pratique, se généraliser à d'autres variables qualitatives codées numériquement. Les variables discriminantes n'ont plus guère de sens mais, si la discrimination fonctionne...

```
proc stepdisc data=sasuser.vispremt;
class carvp;
var sexe ager relat opgnbl--dnbjdl;
run;
```

Les variables ainsi sélectionnées sont utilisées dans deux algorithmes de discrimination. Le premier, non-paramétrique, utilise les k plus proches voisins tandis que le deuxième fait implicitement l'hypothèse de normalité des distributions. Dans les deux cas, une pseudo procédure de validation croisée permet d'estimer les taux de mauvais classement. Il ne s'agit en effet pas d'une procédure de validation croisée explicite car les matrices de variances sont calculées une fois pour toute et donc dépendent des individus à prévoir.

```
proc discrim data= sasuser.vispremt
  method=npar k=11 crossvalidate;
  class CARVP;
  var MOYRVL SEXER BOPPNL GAGECL OPGNBL QCREDL
  FACANL XLGMTL RELAT HAVEFL GAGEML ENDETLL LGAGTL
  VIEMTL TAVEPL ITAVCL AGER;
run;
```

Error Count Estimates for CARVP:

	Cnon	Coui	Total
Rate	0.2191	0.2801	0.2496
Priors	0.5000	0.5000	

```
proc discrim data= sasuser.vispremt
  method=NORMAL crossvalidate;
  class CARVP;
  var MOYRVL SEXER BOPPNL GAGECL OPGNBL QCREDL
```

```
FACANL XLGRTL RELAT HAVEFL GAGEML ENDETL LGAGTL
VIEMTL TAVEPL ITAVCL AGER;
run;
```

	Error Count Estimates for CARVP:		
	Cnon	Coui	Total
Rate	0.1784	0.2689	0.2236
Priors	0.5000	0.5000	

La valeur de k pourrait être sans doute améliorée mais il semble que dans ce cas, l'approche paramétrique fasse un peu mieux. La comparaison entre régression logistique et analyse discriminante demande un peu d'attention et surtout la constitution préalable d'un échantillon test.

Chapitre 5

Arbres binaires

1 Introduction

Ce chapitre s'intéresse aux méthodes ayant pour objectif la construction d'*arbres binaires*, ou dendogrammes, modélisant une discrimination ou une régression. Complémentaires des méthodes statistiques plus classiques : analyse discriminante, régression linéaire, les solutions obtenues sont présentées sous une forme graphique simple à interpréter, même pour des néophytes, et constituent une aide efficace pour l'aide à la décision. Elles sont basées sur un découpage, par des hyperplans, de l'espace engendré par les variables explicatives. Nommées initialement partitionnement récursif ou segmentation, les développements importants de Breiman et col. (1984) les ont fait connaître sous l'acronyme de CART : Classification and Regression Tree ou encore de C4.5 (Quinlan, 1993) dans la communauté informatique. L'acronyme correspond à deux situations bien distinctes selon que la variable à expliquer, modéliser ou prévoir est qualitative (discrimination ou en anglais "classification") ou quantitative (régression).

Ces méthodes ne sont efficaces que pour des tailles d'échantillons importantes et elles sont très calculatoires. Les deux raisons : modèle graphique de décision simple à interpréter, puissance de calcul nécessaire, suffisent à expliquer leur popularité récente. De plus, elles requièrent plutôt moins d'hypothèses que des méthodes statistiques classiques et semblent particulièrement adaptées au cas où les variables explicatives sont nombreuses. En effet, la procédure de sélection des variables est intégrée à l'algorithme construisant l'arbre, d'autre part, les interactions sont prises en compte. Néanmoins, cet algorithme suivant une stratégie pas à pas hiérarchisée, il peut, comme dans le cas du choix de modèle en régression, passer à côté d'un optimum global. Ceci souligne encore l'importance de confronter plusieurs approches sur les mêmes données.

2 Construction d'un arbre binaire

2.1 Principe

Les données sont constituées de l'observation de p variables quantitatives ou qualitatives explicatives X^j et d'une variable à expliquer Y qualitative à m modalités $\{\mathcal{T}_\ell; \ell = 1 \dots, m\}$ ou quantitative réelle, observées sur un échantillon de n individus.

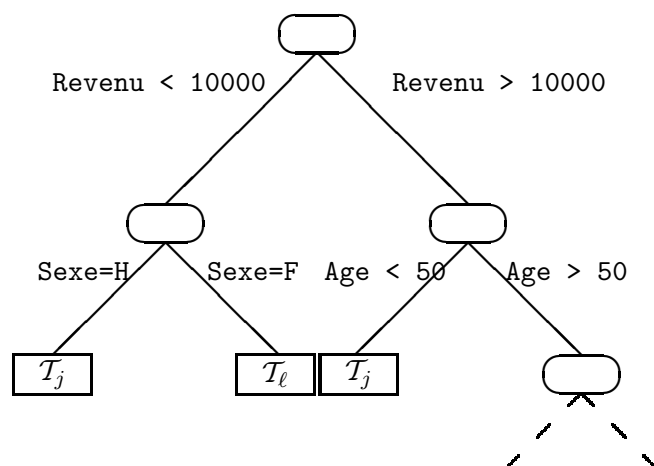


FIG. 5.1 – Exemple élémentaire d’arbre de classification.

La construction d’un arbre de discrimination binaire (cf. figure 2.1) consiste à déterminer une séquence de *nœuds*.

- Un nœud est défini par le choix conjoint d’une variable parmi les explicatives et d’une *division* qui induit une partition en deux classes. Implicitement, à chaque nœud correspond donc un sous-ensemble de l’échantillon auquel est appliquée une dichotomie.
- Une division est elle-même définie par une valeur seuil de la variable quantitative sélectionnée ou un partage en deux groupes des modalités si la variable est qualitative.
- À la racine ou nœud initial correspond l’ensemble de l’échantillon ; la procédure est ensuite itérée sur chacun des sous-ensembles.

L’algorithme considéré nécessite :

- i. la définition d’un critère permettant de sélectionner la “meilleure” division parmi toutes celles *admissibles* pour les différentes variables ;
- ii. une règle permettant de décider qu’un nœud est terminal : il devient ainsi une *feuille* ;
- iii. l’affectation de chaque feuille à l’une des classes ou à une valeur de la variable à expliquer.

Le point (ii) est le plus délicat. Il correspond encore à la recherche d’un modèle parcimonieux. Un arbre trop détaillé, associé à une surparamétrisation, est instable et donc probablement plus défaillant pour la prévision d’autres observations. La contribution majeure de Breiman et col. (1984) est justement une stratégie de recherche d’arbre optimal. Elle consiste à

- i. construire l’arbre maximal A_{\max} ,
- ii. ordonner les sous-arbres selon une séquence emboîtée suivant la décroissance d’un critère pénalisé de déviance ou de taux de mal-classés,
- iii. puis à sélectionner le sous-arbre optimal ; c’est la procédure d’*élagage*.

Tous ces points sont détaillés ci-dessous.

2.2 Critère de division

Une division est dite *admissible* si aucun des deux nœuds descendants qui en découlent n'est vide. Si la variable explicative est qualitative ordinale avec m modalités, elle fournit $(m-1)$ divisions binaires admissibles. Si elle est seulement nominale le nombre de divisions passe à $2^{(m-1)} - 1$. Une variable quantitative se ramène au cas ordinal.

Le critère de division repose sur la définition d'une fonction d'hétérogénéité ou de désordre explicitée dans la section suivante. L'objectif étant de partager les individus en deux groupes les plus homogènes au sens de la variable à expliquer. L'hétérogénéité d'un nœud se mesure par une fonction non négative qui doit être

- i. nulle si, et seulement si, le nœud est homogène : tous les individus appartiennent à la même modalité ou prennent la même valeur de Y .
- ii. Maximale lorsque les valeurs de Y sont équiprobables ou très dispersées.

La division du nœud k crée deux fils, gauche et droit. Pour simplifier, ils sont notés $(k+1)$ et $(k+2)$ mais une re-numérotation est nécessaire pour respecter la séquence de sous-arbres qui sera décrite dans la section suivante.

Parmi toutes les divisions admissibles du nœud k , l'algorithme retient celle qui rend la somme $D_{(k+1)} + D_{(k+2)}$ des désordres des nœuds fils minimales. Ceci revient encore à résoudre à chaque étape k de construction de l'arbre :

$$\max_{\{divisions\ de\ X^j; j=1,p\}} D_k - (D_{(k+1)} + D_{(k+2)})$$

Graphiquement, la longueur de chaque branche peut être représentée proportionnellement à la réduction de l'hétérogénéité occasionnée par la division.

2.3 Règle d'arrêt

La croissance de l'arbre s'arrête à un nœud donné, qui devient donc terminal ou *feuille*, lorsqu'il est homogène c'est-à-dire lorsqu'il n'existe plus de partition admissible ou, pour éviter un découpage inutilement fin, si le nombre d'observations qu'il contient est inférieur à une valeur seuil à choisir en général entre 1 et 5.

2.4 Affectation

Dans le cas Y quantitative, à chaque feuille est associée une valeur : la moyenne des observations associées à cette feuille. Dans le cas qualitatif, chaque feuille ou nœud terminal est affecté à une classe \mathcal{T}_ℓ de Y en considérant le mode conditionnel :

- celle la mieux représentée dans le nœud et il est ensuite facile de compter le nombre d'objets mal classés ;
- la classe *a posteriori* la plus probable au sens bayésien si des probabilités *a priori* sont connues ;
- la classe la moins coûteuse si des coûts de mauvais classement sont donnés.

3 Critères d'homogénéité

Deux cas sont à considérer.

3.1 Y quantitative

On considère le cas plus général d'une division en J classes. Soit n individus et une partition en J classes de tailles $n_j; j = 1, \dots, J$ avec $n = \sum_{j=1}^J n_j$. On numérote $i = 1, \dots, n_j$ les individus de la j ème classe. Soit μ_{ij} (resp. y_{ij}) la valeur "théorique" (resp. l'observation) de Y sur l'individu (i, j) : le i ème de la j ème classe. L'hétérogénéité de la classe j est définie par :

$$D_j = \sum_{i=1}^{n_j} (\mu_{ij} - \mu_{.j})^2 \quad \text{avec} \quad \mu_{.j} = \frac{1}{n_j} \sum_{i=1}^{n_j} \mu_{ij}.$$

L'hétérogénéité de la partition est définie par :

$$D = \sum_{j=1}^J D_j = \sum_{j=1}^J \sum_{i=1}^{n_j} (\mu_{ij} - \mu_{.j})^2;$$

c'est l'inertie intra (homogène à la variance intraclasse) qui vaut $D = 0$ si et seulement si $\mu_{ij} = \mu_{.j}$ pour tout i et tout j .

La différence d'hétérogénéité entre l'ensemble non partagé et l'ensemble partagé selon la partition J est

$$\begin{aligned} \Delta &= \sum_{j=1}^J \sum_{i=1}^{n_j} (\mu_{ij} - \mu_{..})^2 - \sum_{j=1}^J \sum_{i=1}^{n_j} (\mu_{ij} - \mu_{.j})^2 \quad \text{où} \quad \mu_{..} = \frac{1}{n} \sum_{j=1}^J \sum_{i=1}^{n_j} \mu_{ij}. \\ &= \sum_{j=1}^J n_j (\mu_{..} - \mu_{.j})^2; \end{aligned}$$

c'est encore homogène à la variance inter classe ou "désordre" des barycentres qui vaut $\Delta = n_1 n_2 ((\mu_{.1} - \mu_{.2})^2)$ pour $J = 2$ dans le cas qui nous intéresse.

L'objectif, à chaque étape, est de maximiser Δ c'est-à-dire de trouver la variable induisant une partition en 2 classes associée à une inertie (variance) intraclasse minimale ou encore qui rend l'inertie (la variance) interclasse la plus grande.

Les quantités sont estimées :

$$D_j \quad \text{par} \quad \widehat{D}_j = \sum_{i=1}^{n_j} (y_{ij} - y_{.j})^2 \quad (1)$$

$$D \quad \text{par} \quad \widehat{D} = \sum_{j=1}^J \widehat{D}_j = \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - y_{.j})^2. \quad (2)$$

Sous hypothèse gaussienne :

$$Y_{ij} = \mu_{.j} + u_{ij} \quad \text{avec} \quad u_{ij} \sim \mathcal{N}(0, \sigma^2),$$

la log-vraisemblance

$$\log L = \text{Cste} - \frac{n}{2} \log(\sigma^2) - \frac{1}{2\sigma^2} \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - \mu_{.j})^2$$

est rendue maximale pour

$$\mathcal{L}_\mu = \sup_{\mu} \log L = \text{Cste} - \frac{n}{2} \log(\sigma^2) - \frac{1}{2\sigma^2} \sum_{j=1}^J \sum_{i=1}^{n_j} (y_{ij} - y_{.j})^2.$$

Pour le modèle saturé (une classe par individu) : $y_{ij} = \mu_{ij} + u_{ij}$, cet optimum devient :

$$\mathcal{L}_s = \sup_{\mu} \log L = \text{Cste} - \frac{n}{2} \log(\sigma^2).$$

La déviance (par rapport au modèle saturé) s'exprime alors comme :

$$\mathcal{D}_\mu = 2\sigma^2(\mathcal{L}_s - \mathcal{L}_\mu) = \hat{D}.$$

Le raffinement de l'arbre est donc associé à une décroissance, la plus rapide possible, de la déviance. C'est l'optique retenue dans le logiciel Splus. On peut encore dire que la division retenue est celle qui rend le test de Fisher (analyse de variance), comparant les moyennes entre les deux classes, le plus significatif possible.

3.2 Y qualitative

Dans ce cas, la fonction d'hétérogénéité, ou de désordre d'un nœud, est définie à partir de la notion d'entropie, du critère de concentration de Gini ou encore d'une statistique de test du χ^2 . En pratique, il s'avère que le choix du critère importe moins que celui du niveau d'élagage. Le premier critère (entropie) est souvent préféré (Splus) car il s'interprète encore comme un terme de déviance mais d'un modèle multinomial cette fois.

On considère une variable à expliquer qualitative, Y à m modalités ou catégories \mathcal{T} numérotées $\ell = 1, \dots, m$. L'arbre induit une partition pour laquelle n_{+k} désigne l'effectif de la k ème classe ou k ème nœud. Soit

$$p_{\ell k} = P[\mathcal{T}_\ell \mid k] \quad \text{avec} \quad \sum_{\ell=1}^m p_{\ell k} = 1$$

la probabilité qu'un élément du k ème nœud appartienne à la ℓ ème classe.

Le *désordre* du k ème nœud, défini à partir de l'entropie, s'écrit avec la convention $0 \log(0) = 0$:

$$D_k = -2 \sum_{\ell=1}^m n_{+k} p_{\ell k} \log(p_{\ell k})$$

tandis que l'hétérogénéité ou désordre de la partition est encore :

$$D = \sum_{k=1}^K D_k = -2 \sum_{k=1}^K \sum_{\ell=1}^m n_{+k} p_{\ell k} \log(p_{\ell k}).$$

Cette quantité est positive ou nulle, elle est nulle si et seulement si les probabilités $p_{\ell k}$ ne prennent que des valeurs 0 sauf une égale à 1 correspondant à l'absence de mélange.

Désignons par $n_{\ell k}$ l'effectif observé de la ℓ ème classe dans le k ème nœud. Un nœud k de l'arbre représente un sous-ensemble de l'échantillon d'effectif $n_{+k} = \sum_{\ell=1}^m n_{\ell k}$.

Les quantités sont estimées :

$$D_k \text{ par } \widehat{D}_k = -2 \sum_{\ell=1}^m n_{+k} \frac{n_{\ell k}}{n_{+k}} \log \frac{n_{\ell k}}{n_{+k}} \quad (3)$$

$$D \text{ par } \widehat{D} = \sum_{k=1}^K \widehat{D}_k = -2 \sum_{k=1}^K \sum_{\ell=1}^m n_{\ell k} \log \frac{n_{\ell k}}{n_{+k}}. \quad (4)$$

Considérons, pour chaque classe ou nœud k , un modèle multinomial à m catégories de paramètre :

$$p_k = (p_{1k}, \dots, p_{mk}), \quad \text{avec} \quad \sum_{\ell=1}^m p_{\ell k} = 1.$$

Pour ce modèle, la logvraisemblance :

$$\log L = \text{Cste} + \sum_{k=1}^K \sum_{\ell=1}^m n_{\ell k} \log(p_{\ell k})$$

est rendue maximale pour

$$\mathcal{L}_\mu = \sup_{p_{\ell k}} \log L = \text{Cste} + \sum_{k=1}^K \sum_{\ell=1}^m n_{\ell k} \log \frac{n_{\ell k}}{n_{+k}}.$$

Pour le modèle saturé (une catégorie par objet), cet optimum prend la valeur de la constante et la déviance (par rapport au modèle saturé) s'exprime comme :

$$\mathcal{D} = -2 \sum_{k=1}^K \sum_{\ell=1}^m n_{\ell k} \log \frac{n_{\ell k}}{n_{+k}} = \widehat{D}.$$

Comme pour l'analyse discriminante décisionnelle, les probabilités conditionnelles sont définies par la règle de Bayes lorsque les probabilités *a priori* π_ℓ d'appartenance à la ℓ ème classe sont connues. Dans le cas contraire, les probabilités de chaque classe sont estimées sur l'échantillon et donc les probabilités conditionnelles s'estiment simplement par des rapports d'effectifs : $p_{\ell k}$ est estimée par $n_{\ell k}/n_{+k}$. Enfin, il est toujours possible d'introduire, lorsqu'ils sont connus, des coûts de mauvais classement et donc de se ramener à la minimisation d'un risque bayésien.

4 Élagage

Dans des situations complexes, la démarche proposée conduit à des arbres extrêmement raffinés et donc à des modèles de prévision très instables car fortement dépendants des échantillons qui ont permis leur estimation. On se trouve donc dans une situation de sur-ajustement à éviter au profit de modèles plus parcimonieux donc plus robuste au moment de la prévision. Cet objectif est obtenu par une procédure d'*élagage* (*pruning*) de l'arbre.

Le principe de la démarche, introduite par Breiman et col. (1984), consiste à construire une suite emboîtée de sous-arbres de l'arbre maximum par élagage successif puis à choisir, parmi cette suite, l'arbre optimal au sens d'un critère. La solution ainsi obtenue par un algorithme pas à pas n'est pas nécessairement globalement optimale mais l'efficacité et la fiabilité sont préférées à l'optimalité.

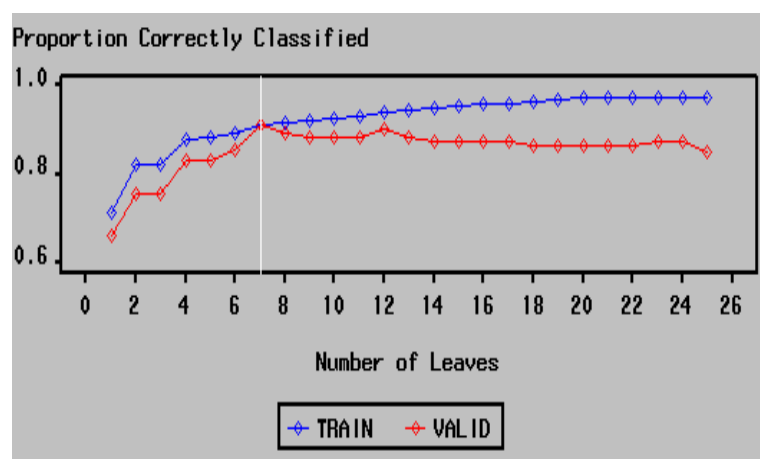


FIG. 5.2 – Carte Visa : choix du nombre de feuilles par échantillon de validation (SEM, 2001).

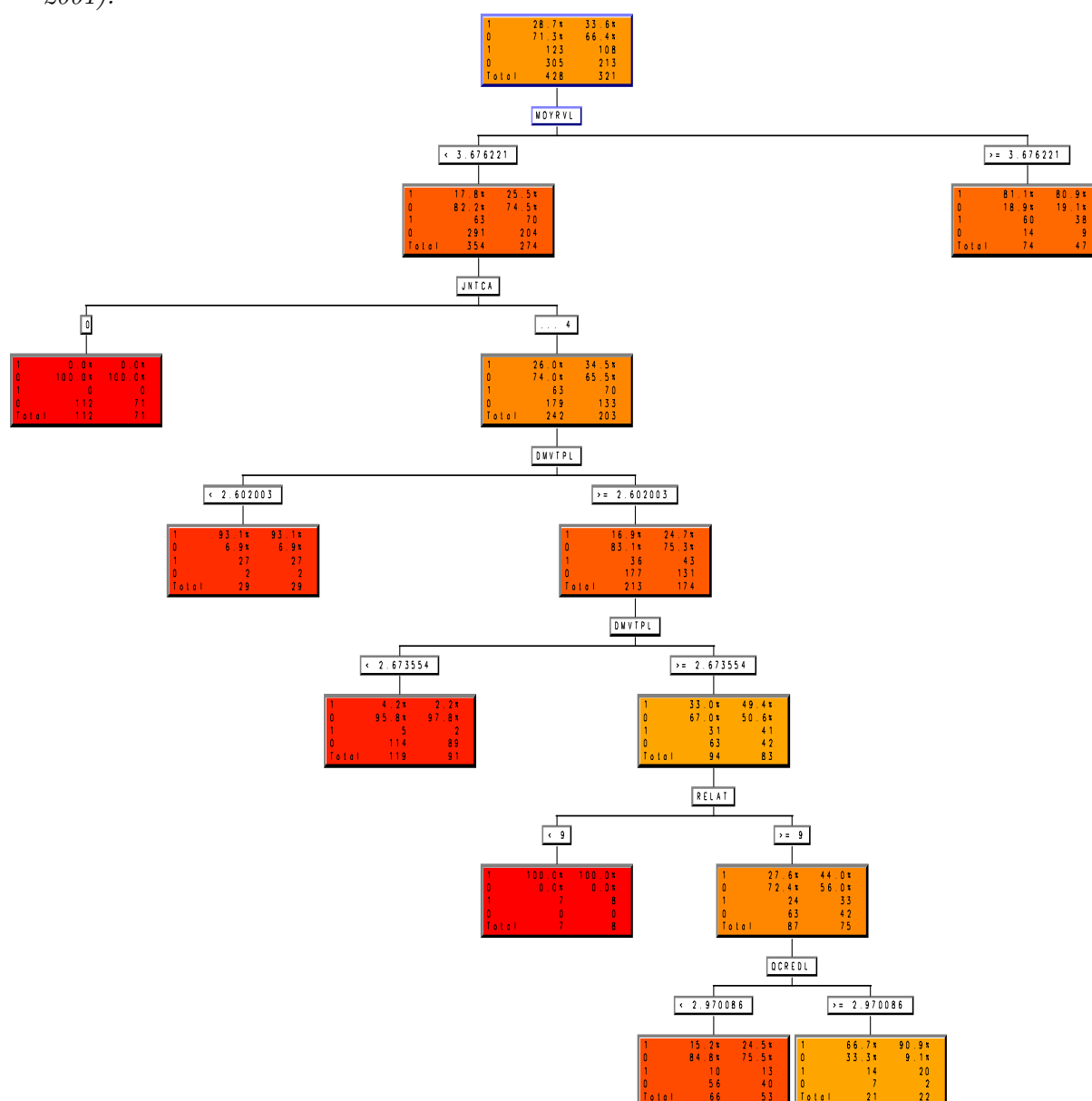


FIG. 5.3 – Carte Visa : arbre de décision élagué suivant l'échantillon de validation (SEM, 2001).

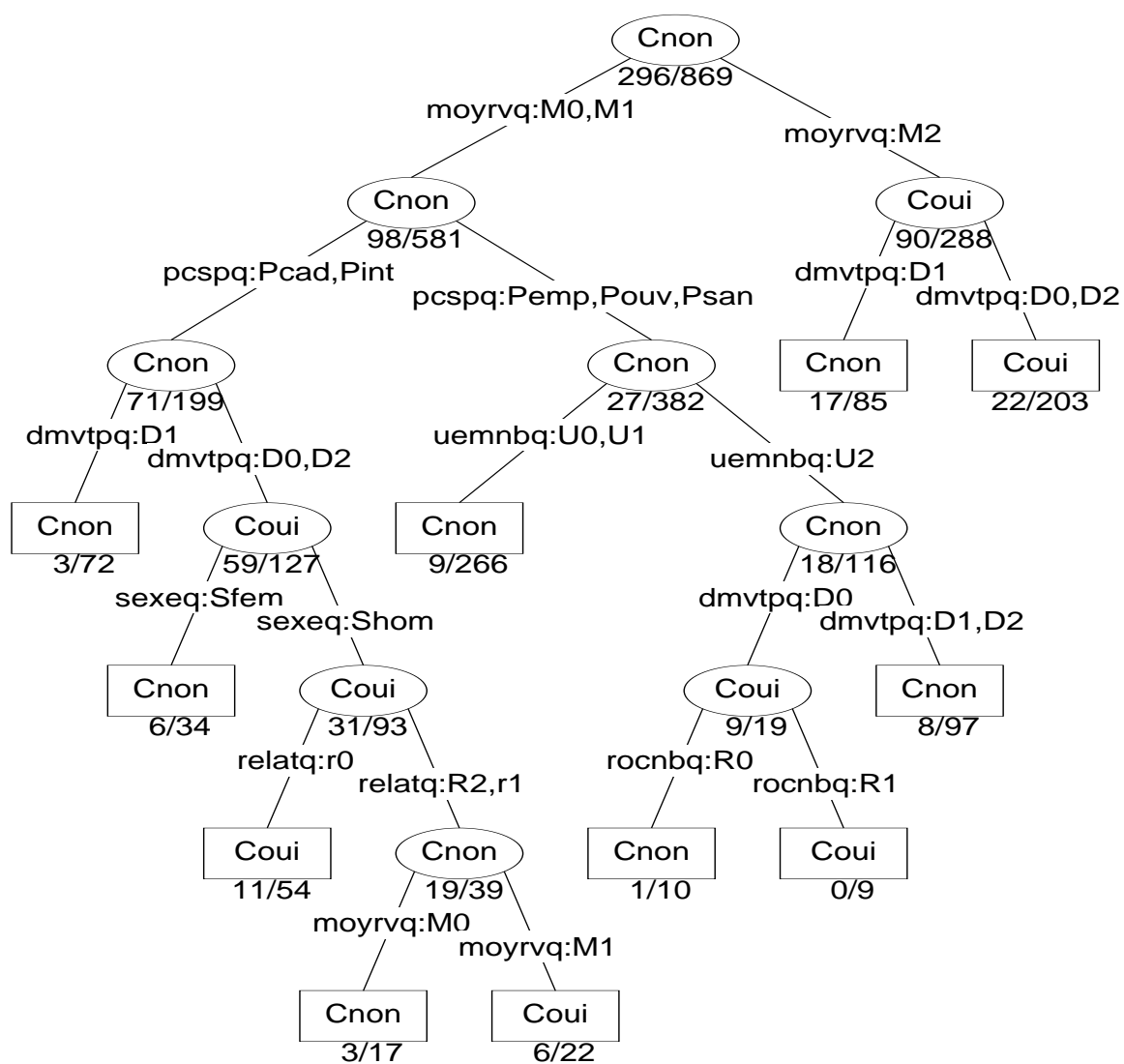


FIG. 5.4 – Carte Visa : arbre de décision (Splus, 1993) élagué par validation croisée.

4.1 Construction de la séquence d'arbres

Pour un arbre A donné, on note K le nombre de feuilles ou nœuds terminaux de A ; la valeur de K exprime la complexité de A . La mesure de qualité de discrimination d'un arbre A s'exprime par un critère

$$D(A) = \sum_{k=1}^K D_k(A)$$

où $D_k(A)$ est le nombre de mal classés ou la déviance ou le coût de mauvais classement de la k ème feuille de l'arbre A .

La construction de la séquence d'arbres emboîtés repose sur une pénalisation de la complexité de l'arbre :

$$C(A) = D(A) + \gamma K.$$

Pour $\gamma = 0$, $A_{\max} = A_K$ minimise $C(A)$. En faisant croître γ , l'une des divisions de A_K , celle pour laquelle l'amélioration de D est la plus faible (inférieure à γ), apparaît comme superflue et les deux feuilles obtenues sont regroupées (élaguées) dans le nœud père qui devient terminal; A_K devient A_{K-1} .

Le procédé est itéré pour la construction de la séquence emboîtée :

$$A_{\max} = A_K \supset A_{K-1} \supset \cdots A_1$$

où A_1 , le nœud racine, regroupe l'ensemble de l'échantillon.

Un graphe représente la décroissance ou éboulis de la déviance (ou du taux de mal classés) en fonction du nombre croissant de feuilles dans l'arbre ou, c'est équivalent, en fonction de la valeur décroissante du coefficient de pénalisation γ .

4.2 Recherche de l'arbre optimal

Les procédures d'élagage diffèrent par la façon d'estimer l'erreur de prédiction. Le graphe précédemment obtenu peut se lire comme un éboulis de valeur propre. Quand l'amélioration du critère est jugé trop petite ou négligeable, on élague l'arbre au nombre de feuilles obtenues. L'évaluation de la déviance ou du taux de mauvais classement estimée par resubstitution sur l'échantillon d'apprentissage est biaisée (trop optimiste). Une estimation sans biais est obtenue par l'utilisation d'un autre échantillon (validation) ou encore par validation croisée. La procédure de validation croisée présente dans ce cas une particularité car la séquence d'arbres obtenue est différente pour chaque estimation sur l'un des sous-échantillons. L'erreur moyenne n'est pas, dans ce cas, calculée pour chaque sous-arbre avec un nombre de feuilles donné mais pour chaque sous-arbre correspondant à une valeur fixée du coefficient de pénalisation. À la valeur de γ minimisant l'estimation de l'erreur de prévision, correspond ensuite l'arbre jugé optimal dans la séquence estimée sur tout l'échantillon d'apprentissage.

Le principe de sélection d'un arbre optimal est donc décrit dans l'algorithme ci-dessous.

- Construction de l'arbre maximal A_{max} .
 - Construction de la séquence $A_K \dots A_1$ d'arbres emboîtés.
 - Estimation sans biais (échantillon de validation ou validation croisée) des déviations $D(A_K), \dots, D(A_1)$.
 - Représentation de $D(A_k)$ en fonction de k ou de γ .
 - Choix de k rendant $D(A_k)$ minimum.
-

Chapitre 6

Méthodes connexionistes

1 Historique

Nous nous intéressons ici à une branche de l'Informatique fondamentale qui, sous l'appellation d'*Intelligence Artificielle*, a pour objectif de simuler des comportements du cerveau humain. Les premières tentatives de modélisation du cerveau sont anciennes et précèdent même l'ère informatique. C'est en 1943 que Mc Culloch (neurophysiologiste) et Pitts (logicien) ont proposé les premières notions de *neurone formel*. Ce concept fut ensuite mis en réseau avec une couche d'entrée et une sortie par Rosenblatt en 1959 pour simuler le fonctionnement rétinien et tenter de reconnaître des formes. C'est l'origine du *perceptron*. Cette approche dite *connexioniste* a atteint ses limites technologiques, compte tenu de la puissance de calcul de l'époque, mais aussi théoriques au début des années 70.

L'approche connexioniste à *connaissance répartie* a alors été supplantée par l'approche *symbolique* ou séquentielle qui promouvait les *systèmes experts* à *connaissance localisée*. L'objectif était alors d'automatiser le principe de l'expertise humaine en associant trois concepts :

- une *base de connaissance* dans laquelle étaient regroupées “toutes” les connaissances d'experts humains sous forme de propositions logiques élémentaires ou plus élaborées en utilisant des quantificateurs (logique du premier ordre).
- une *base de faits* contenant les observations du cas à traiter comme, par exemple, des résultats d'examens, d'analyses de sang, de salive pour des applications biomédicales de choix d'un antibiotique,
- un *moteur d'inférence* chargé d'appliquer les règles expertes sur la base de faits afin d'en déduire de nouveaux faits jusqu'à la réalisation d'un objectif comme l'élaboration du traitement d'une infection bactérienne.

Face aux difficultés rencontrées lors de la modélisation des connaissances d'un expert humain, au volume considérable des bases de connaissance qui en découlait et au caractère exponentiel de la complexité des algorithmes d'inférence mis en jeu, cette approche s'est éteinte avec les années 80. En effet, pour les systèmes les plus compliqués à base de calcul des prédicats du premier ordre, on a pu montrer qu'ils conduisaient à des problèmes *NP* complets et donc dont la solution pouvait être atteinte mais pas nécessairement en un temps fini !

L'essor technologique et surtout quelques avancées théoriques :

- algorithme d'estimation par rétropropagation de l'erreur par Hopkins en 1982,

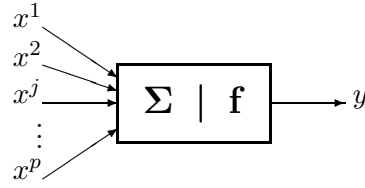


FIG. 6.1 – Représentation d'un neurone formel.

• analogie de la phase d'apprentissage avec les modèles markoviens de systèmes de particules de la mécanique statistique (verres de spin) par Hopfield en 1982, au début des années 80 ont permis de relancer l'approche connexioniste. Celle-ci a connu au début des années 90 un développement considérable si l'on considère le nombre de publications et de congrès qui lui ont été consacrés mais aussi les domaines d'applications très divers où elle apparaît. Sur de nombreux objectifs, justement ceux propres au data mining, les réseaux neuronaux ne rentrent pas nécessairement en concurrence avec des méthodes statistiques bientôt centenaires mais apportent un point de vue complémentaire qu'il est important de considérer (Thiria et col. 1997).

2 Réseaux de neurones

Un *réseau neuronal* est l'association, en un graphe plus ou moins complexe, d'objets élémentaires, les *neurones formels*. Les principaux réseaux se distinguent par l'organisation du graphe (en couches, complets...), c'est-à-dire leur architecture, son niveau de complexité (le nombre de neurones) et par le type des neurones (leurs fonctions de transition).

2.1 Neurone formel

De façon très réductrice, un neurone biologique est une cellule qui se caractérise par

- des synapses, les points de connexion avec les autres neurones, fibres nerveuses ou musculaires ;
- des dendrites, les “entrées” du neurones ;
- l'axone, la “sortie” du neurone vers d'autres neurones ou fibres musculaires ;
- le noyau qui active la sortie en fonction des stimuli en entrée.

Par analogie, le neurone formel est un modèle qui se caractérise par un état interne $s \in \mathcal{S}$, des signaux d'entrée x_1, \dots, x_p et une fonction de transition d'état

$$s = h(x_1, \dots, x_p) = f \left(\beta_0 + \sum_{j=1}^p \beta_j x_j \right).$$

La fonction de transition opère une transformation d'une combinaison affine des signaux d'entrée, β_0 étant appelé le biais du neurone. Cette combinaison affine est déterminée par

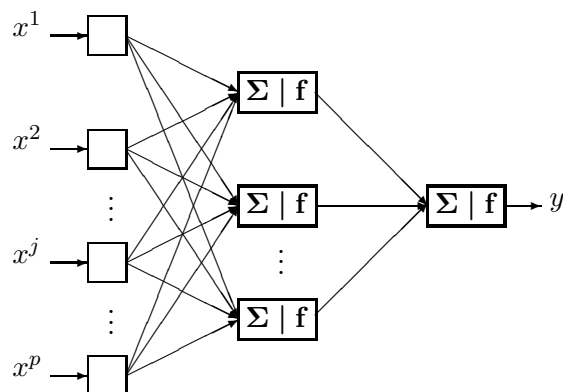


FIG. 6.2 – Exemple de perceptron multicouche élémentaire avec une couche cachée et une couche de sortie.

un *vecteur de poids* $[\beta_0, \dots, \beta_p]$ associé à chaque neurone et dont les valeurs sont estimées dans la phase d'apprentissage. Ils constituent “la mémoire” ou “connaissance répartie” du réseau.

Les différents types de neurones se distinguent par la nature f de leur fonction de transition. Les principaux types sont :

- *linéaire* f est la fonction identité,
- *sigmoïde* $f(x) = 1/(1 + e^x)$,
- *seuil* $f(x) = \mathbf{1}_{[0, +\infty[}(x)$,
- *stochastiques* $f(x) = 1$ avec la probabilité $1/(1 + e^{-x/H})$, 0 sinon (H intervient comme une température dans un algorithme de recuit simulé),
- ...

Les modèles linéaires et sigmoïdaux sont bien adaptés aux algorithmes d'apprentissage comme celui de rétropropagation du gradient car leur fonction de transition est différentiable. Ce sont les plus utilisés. Le modèle à seuil est sans doute plus conforme à la “réalité” biologique mais pose des problèmes d'apprentissage. Enfin le modèle stochastique est utilisé pour des problèmes d'optimisation globale de fonctions perturbées ou encore pour les analogies avec les systèmes de particules. On ne le rencontre pas en data mining.

3 Perceptron multicouche

3.1 Architecture

Le perceptron multicouche (PMC) est un réseau composé de couches successives. Une *couche* est un ensemble de neurones n'ayant pas de connexion entre eux. Une couche d'entrée lit les signaux entrant, un neurone par entrée x_j , une couche en sortie fournit la réponse du système. Selon les auteurs, la couche d'entrée qui n'introduit aucune modifica-

tion n'est pas comptabilisée. Une ou plusieurs couches cachées participent au transfert. Un neurone d'une couche cachée est connecté en entrée à chacun des neurones de la couche précédente et en sortie à chaque neurone de la couche suivante.

Un perceptron multicouche réalise donc une transformation

$$y = F(x_1, \dots, x_p; \beta)$$

où β est le vecteur contenant chacun des paramètres β_{jkl} de la j ème entrée du k ème neurone de la ℓ ème couche; la couche d'entrée ($\ell = 0$) n'est pas paramétrée, elle ne fait que distribuer les entrées sur tous les neurones de la couche suivante.

Par souci de cohérence, nous avons tâché de conserver les mêmes notations à travers les différents chapitres. Ainsi, les *entrées* d'un réseau sont encore notées x^1, \dots, x^p comme les variables explicatives d'un modèle tandis que les *poids* des entrées sont des paramètres β à estimer lors de la procédure d'*apprentissage* et que la *sortie* est la variable à expliquer ou cible du modèle.

3.2 Apprentissage

Supposons que l'on dispose d'une base d'apprentissage de taille n d'observations $(x_i^1, \dots, x_i^p; y_i)$ des variables explicatives X^1, \dots, X^p et de la variable à prévoir Y . L'apprentissage est l'estimation $\hat{\beta}$ des paramètres du modèle solutions du problème des moindres carrés¹ :

$$\hat{\beta} = \arg \min_{\mathbf{b}} Q(\mathbf{b}) \quad \text{avec} \quad Q(\mathbf{b}) = \frac{1}{n} \sum_{i=1}^n [y_i - F(x_i^1, \dots, x_i^p; (b))]^2.$$

L'algorithme d'optimisation le plus utilisé est celui de rétropropagation du gradient basé sur l'idée suivante : en tout point \mathbf{b} , le vecteur gradient de Q pointe dans la direction de l'erreur croissante. Pour faire décroître Q il suffit donc de se déplacer en sens contraire. Il s'agit d'un algorithme itératif modifiant les poids de chaque neurone selon :

$$b_{jkl}(i) = b_{jkl}(i-1) + \Delta b_{jkl}(i)$$

où la correction $\Delta b_{jkl}(i)$ est proportionnelle au gradient et à l'erreur attribuée à l'entrée concernée $\varepsilon_{jkl}(i)$ et incorpore un terme d'"inertie" $\alpha b_{jkl}(i-1)$ permettant d'amortir les oscillations du système :

$$\Delta b_{jkl}(i) = -\tau \varepsilon_{jkl}(i) \frac{\partial Q}{\partial b_{jkl}} + \alpha b_{jkl}(i-1).$$

Le coefficient de proportionnalité τ est appelé le *taux d'apprentissage*. Il peut être fixe à déterminer par l'utilisateur ou encore varier en cours d'exécution selon certaines règles paramétrées par l'utilisateur. Il paraît en effet intuitivement raisonnable que, grand au début pour aller plus vite, ce taux décroisse pour aboutir à un réglage plus fin au fur et à mesure que le système s'approche d'une solution. La formule de *rétropropagation de l'erreur* fournit, à partir des erreurs observées sur les sorties, l'expression de l'erreur attribuée à chaque entrée de la couche de sortie à la couche d'entrée.

La littérature sur le sujet propose quantités de recettes destinées à améliorer la vitesse de convergence de l'algorithme ou bien lui éviter de rester collé à une solution locale

¹Équivalent à une maximisation de la vraisemblance dans le cas gaussien.

défavorable. Des propriétés (dynamique markovienne ergodique et convergence vers la mesure stationnaire) de cet algorithme impliquent une convergence presque sûre ; la probabilité d'atteindre une précision fixée *a priori* tend vers 1 lorsque la taille de l'échantillon d'apprentissage tend vers l'infini.

ALGORITHME 6.1 : Rétropropagation du gradient

- **Initialisation**
 - Les poids b_{jkl} par tirage aléatoire selon une loi uniforme sur $[0, 1]$.
 - Normaliser dans $[0, 1]$ les données d'apprentissage.
 - **Tant que** $Q > err_{max}$ ou $niter < iter_{max}$.
 - **Ranger la base d'apprentissage dans un nouvel ordre aléatoire.**
 - **Pour** chaque élément $i = 1, \dots, n$ de la base **Faire**
 - Calculer $\varepsilon(i) = y_i - F(x_i^1, \dots, x_i^p; (b)(i-1))$ en propageant les entrées vers l'avant.
 - L'erreur est "rétropropagée" dans les différentes couches afin d'affecter à chaque entrée une responsabilité dans l'erreur globale.
 - Mise à jour de chaque poids $b_{jkl}(i) = b_{jkl}(i-1) + \Delta b_{jkl}(i)$
 - **Fin Pour**
 - **Fin Tant que**
-

3.3 Utilisation

On pourra se reporter à l'abondante littérature sur le sujet (Haykin, 1994) pour obtenir des précisions sur les algorithmes d'apprentissage et leurs nombreuses variantes. Il est important de rappeler la liste des choix qui sont laissés à l'utilisateur. En effet, même si les logiciels proposent des valeurs par défaut, il est fréquent que cet algorithme connaisse quelques soucis de convergence.

L'utilisateur doit donc déterminer

- i. les variables d'entrée et la variable de sortie ; leur faire subir comme pour toutes méthodes statistiques, d'éventuelles transformations.
- ii. L'architecture du réseau : le nombre de couches cachées (en général une ou deux) qui correspond à une aptitude à traiter des problèmes de non-linéarité, le nombre de neurones par couche cachée. Ces deux choix conditionnent directement le nombre de paramètres (de poids) à estimer. Ils participent à la recherche d'un bon compromis biais/variance c'est-à-dire à l'équilibre entre qualité d'apprentissage et qualité de prévision. À la louche, on considère en pratique qu'il faut un échantillon d'apprentissage au moins dix fois plus grand que le nombre de paramètres à estimer.
- iii. Deux autres paramètres interviennent également sur ce compromis : le nombre maximum d'itérations et l'erreur maximum tolérée. En renforçant ces critères on améliore la qualité de l'apprentissage ce qui peut se faire au détriment de celle de la prévision.
- iv. Le taux d'apprentissage ainsi qu'une éventuelle stratégie d'évolution de celui-ci.

Le nombre de couches reste restreint. On montre en effet que toute fonction que toute fonction continue d'un compact de \mathbb{R}^P dans \mathbb{R}^q peut être approchée avec une précision arbitraire par un réseau à une couche cachée en adaptant le nombre de neurones. La complexité du modèle est contrôlée par le nombre de neurones ou encore par la durée de

l'apprentissage. Ces paramètres sont optimisés en considérant un échantillon de validation et le plus simple consiste à arrêter l'apprentissage lorsque l'erreur sur l'échantillon de validation commence à se dégrader tandis que celle sur l'échantillon d'apprentissage ne peut que continuer à décroître.

Les champs d'application des PMC sont très nombreux : discrimination, prévision d'une série temporelle, reconnaissance de forme... Ils sont en général bien explicités dans les documentations des logiciels spécialisés.

Les critiques principales énoncées à l'encontre du PMC concernent les difficultés liées à l'apprentissage (temps de calcul, taille de l'échantillon, localité de l'optimum obtenu) ainsi que son statut de boîte noire. En effet, contrairement à un modèle de discrimination ou un arbre, il est *a priori* impossible de connaître l'influence effective d'une entrée (une variable) sur le système dès qu'une couche cachée intervient. Néanmoins, des techniques de recherche de sensibilité du système à chacune des entrées permettent de préciser les idées et, éventuellement de simplifier le système en supprimant certaines des entrées.

En revanche, ils possèdent d'indéniables qualités lorsque l'absence de linéarité et/ou le nombre de variables explicatives rendent les modèles statistiques traditionnelles inutilisables. Leur flexibilité alliée à une procédure d'apprentissage intégrant la pondération (le choix) des variables comme de leurs interactions peuvent les rendre très efficaces (Besse et col. 2001).

Chapitre 7

Agrégation de modèles

1 Introduction

Ce chapitre décrit des algorithmes plus récemment apparus dans la littérature. Ils sont basés sur des stratégies adaptatives (*boosting*) ou aléatoires (*bagging*) permettant d'améliorer l'ajustement par une combinaison ou agrégation d'un grand nombre de modèles tout en évitant un sur-ajustement. Ces algorithmes se sont développés à la frontière entre apprentissage machine (*machine learning*) et Statistique. De nombreux articles comparatifs montrent leur efficacité sur des exemples de données simulées et surtout pour des problèmes réels complexes (voir par exemple Ghattas 2000) tandis que leurs propriétés théoriques sont un thème de recherche actif.

Deux types d'algorithmes sont décrits schématiquement dans ce chapitre. Ceux reposants sur une construction aléatoires d'une famille de modèle : *bagging* pour *bootstrap aggregating* (Breiman 1996), les forêts aléatoires (*random forests*) de Breiman (2001) qui propose une amélioration du *bagging* spécifique aux modèles définis par des arbres binaires (CART). Ceux basés sur le *boosting* (Freund et Shapiro, 1996), reposent sur une construction *adaptive*, déterministe ou aléatoire, d'une famille de modèles.

Les principes du *bagging* ou du *boosting* s'appliquent à toute méthode de modélisation (régression, CART, réseaux de neurones) mais n'ont d'intérêt, et réduisent sensiblement l'erreur de prédiction, que dans le cas de modèles *instables*, donc plutôt non linéaires. Ainsi, l'utilisation de ces algorithmes n'a guère de sens avec la régression multilinéaire ou l'analyse discriminante. Ils sont surtout mises en œuvre en association avec des arbres binaires comme modèles de base.

2 Famille de modèles aléatoires

2.1 *Bagging*

Principe et algorithme

Soit Y une variable à expliquer quantitative ou qualitative, X^1, \dots, X^p les variables explicatives et $\phi(\mathbf{x})$ un modèle fonction de $\mathbf{x} = \{x^1, \dots, x^p\} \in \mathbb{R}^p$. On note n le nombre

d'observations et

$$\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$$

un échantillon de loi F .

L'espérance $\phi(\cdot) = E_F(\hat{\phi}_{\mathbf{z}})$ de l'estimateur définie sur l'échantillon \mathbf{z} , est un estimateur sans biais de variance nulle. Considérons B échantillons indépendants notés $\{\mathbf{z}_b\}_{b=1,B}$ et construisons une agrégation des modèles dans le cas où la variable à expliquer Y est :

- quantitative : $\hat{\phi}_B(\cdot) = \frac{1}{B} \sum_{b=1}^B \hat{\phi}_{\mathbf{z}_b}(\cdot)$,
- qualitative : $\hat{\phi}_B(\cdot) = \arg \max_j \text{card} \{b \mid \hat{\phi}_{\mathbf{z}_b}(\cdot) = j\}$.

Dans le premier cas, il s'agit d'une simple moyenne des résultats obtenus pour les modèles associés à chaque échantillon, dans le deuxième, un *comité* de modèles est constitué pour voter et élire la réponse la plus probable. Dans ce dernier cas, si le modèle retourne des probabilités associées à chaque modalité comme en régression logistique ou avec les arbres de décision, il est aussi simple de calculer des moyennes de ces probabilités.

Le principe est élémentaire, moyenner les prédictions de plusieurs modèles indépendants permet de réduire la variance et donc de réduire l'erreur de prédiction.

Cependant, il n'est pas réaliste de considérer B échantillons indépendants. Cela nécessiterait généralement trop de données. Ces échantillons sont donc remplacés par B répliques d'échantillons *bootstrap* (cf. Annexe B) obtenus chacun par n tirages avec remise selon la mesure empirique \hat{F} . Ceci conduit à l'algorithme ci-dessous.

ALGORITHME 7.1 : Bagging

- Soit \mathbf{x}_0 à prévoir et
 - $\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ un échantillon
 - Pour $b = 1$ à B Faire
 - Tirer un échantillon bootstrap \mathbf{z}_b^* .
 - Estimer $\hat{\phi}_{\mathbf{z}_b}(\mathbf{x}_0)$ sur l'échantillon bootstrap.
 - Fin Pour

Calculer l'estimation moyenne $\hat{\phi}_B(\mathbf{x}_0) = \frac{1}{B} \sum_{b=1}^B \hat{\phi}_{\mathbf{z}_b}(\mathbf{x}_0)$ ou le résultat du vote.
-

Utilisation

Il est naturel et techniquement facile d'accompagner ce calcul par une estimation *bootstrap out-of-bag* (cf. chapitre 3 section 4.2) de l'erreur de prédiction. Elle est une mesure de la qualité de généralisation du modèle et permet de prévenir une éventuelle tendance au surajustement. C'est, pour éviter un biais, la moyenne des erreurs de prédiction commises par chaque estimateur ; chacune des erreurs étant estimée sur les observations qui n'ont pas été sélectionnées par l'échantillon *bootstrap* correspondant.

En pratique, CART est souvent utilisée comme méthode de base pour construire une famille de modèles c'est-à-dire d'arbres binaires. Trois stratégies d'élagage sont alors possibles :

- i. laisser construire et garder un arbre complet pour chacun des échantillons,
- ii. construire un arbre d'au plus q feuilles,
- iii. construire à chaque fois l'arbre complet puis l'élaguer par validation croisée.

La première stratégie semble en pratique un bon compromis entre volume des calculs et qualité de prédiction. Chaque arbre est alors affecté d'un faible biais et d'une grande variance mais la moyenne des arbres réduit avantageusement celle-ci. En revanche, l'élagage par validation croisée pénalise lourdement les calculs sans gain substantiel de qualité.

Cet algorithme a l'avantage de la simplicité, il s'adapte et se programme facilement quelque soit la méthode de modélisation mise en œuvre. Il pose néanmoins quelques problèmes :

- temps de calcul important pour évaluer un nombre suffisant d'arbres jusqu'à ce que l'erreur de prédiction *out-of-bag* ou sur un échantillon validation se stabilise et arrête si elle tend à augmenter ;
- nécessiter de stocker tous les modèles de la combinaison afin de pouvoir utiliser cet outil de prédiction sur d'autres données,
- l'amélioration de la qualité de prédiction se fait au détriment de l'interprétabilité. Le modèle finalement obtenu devient une *boîte noire* comme dans le cas du perceptron.

2.2 Forêts aléatoires

Algorithme

Dans le cas spécifique des modèles CART (arbres binaires), Breiman (2001) propose une amélioration du *bagging* par l'ajout d'une randomisation. L'objectif est donc de rendre plus *indépendants* les arbres de l'agrégation en ajoutant du hasard dans le choix des variables qui interviennent dans les modèles. Cette approche semble plus particulièrement fructueuse dans des situations hautement multidimensionnelles, c'est-à-dire lorsque le nombre de variables explicatives p est très important. C'est le cas lorsqu'il s'agit, par exemple, de discriminer des courbes, spectres, signaux, biopuces.

ALGORITHME 7.2 : Forêts aléatoires

- Soit \mathbf{x}_0 à prévoir et
 - $\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ un échantillon
 - Pour $b = 1$ à B Faire
 - Tirer un échantillon bootstrap \mathbf{z}_b^*
 - Estimer un arbre sur cet échantillon avec randomisation des variables selon l'une des deux options :
 - i. Si le nombre de variables explicatives est important, la recherche de chaque nœud optimal est précédé d'un tirage aléatoire d'un sous-ensemble de q prédicteurs.
 - ii. Sinon, tirer au hasard $q_1 \approx 3$ variables explicatives puis construire q_2 "prédicteurs" par combinaisons linéaires avec des coefficients obtenus par tirages aléatoires uniformes sur $[0, 1]$.
 - Fin Pour
- Calculer l'estimation moyenne $\hat{\phi}_B(\mathbf{x}_0) = \frac{1}{B} \sum_{b=1}^B \hat{\phi}_{\mathbf{z}_b}(\mathbf{x}_0)$ ou le résultat du vote.
-

Élagage

La stratégie d'élagage peut, dans le cas des forêts aléatoires, être plus élémentaire qu'avec le *bagging* en se limitant à des arbres de taille q relativement réduite voire même tri-

viale avec $q = 2$ (*stump*). En effet, avec le seul *bagging*, des arbres limités à une seule fourche risquent d'être très semblables (fortement corrélés) car impliquant les mêmes quelques variables apparaissant comme les plus explicatives. La sélection aléatoire d'un nombre réduit de prédicteurs potentiels à chaque étape de construction d'un arbre, accroît significativement la variabilité en mettant en avant nécessairement d'autres variables. Chaque modèle de base est évidemment moins performant mais, l'union faisant la force, l'agrégation conduit finalement à de bons résultats. Le nombre de variables tirées aléatoirement n'est pas un paramètre sensible un choix par défaut de $q = \sqrt{p}$ est suggéré par Breiman (2001). Comme pour le *bagging*, l'évaluation itérative de l'erreur *out-of-bag* prévient d'un éventuel surajustement si celle-ci vient à se dégrader.

L'interprétation est ensuite facilitée par le calcul et la représentation graphique d'un indice proportionnel à l'importance de chaque variable dans l'agrégation de modèles et donc de sa participation à la régression ou à la discrimination. C'est évidemment d'autant plus utile que les variables sont très nombreuses. Plusieurs critères sont proposés par Breiman (2001) pour évaluer l'importance de la j ème variable. Ils reposent sur une permutation aléatoire des valeurs de cette variable. L'un de ces critères consiste à calculer la moyenne sur toutes les observations de la décroissance de leur marge lorsque la variable est aléatoirement perturbée. La marge est ici la proportion de votes pour la vraie classe d'une observation moins le maximum des proportions des votes pour les autres classes.

3 Famille de modèles adaptatifs

3.1 Principes du *Boosting*

Le *boosting* diffère des approches précédentes par ses origines et ses principes. L'idée initiale, en apprentissage machine, était d'améliorer les compétences d'un *faible classifieur* c'est-à-dire celle d'un modèle de discrimination dont la probabilité de succès sur la prédiction d'une variable qualitative est légèrement supérieure à celle d'un choix aléatoire. L'idée originale de Schapire (1990) a été affinée par Freund et Schapire (1996) qui ont décrit l'algorithme original *AdaBoost* (*Adaptive boosting*) pour la prédiction d'une variable binaire. De nombreuses études ont ensuite été publiées pour adapter cet algorithme à d'autres situations : k classes, régression et rendre compte de ses performances sur différents jeux de données (cf. Schapire, 2002) pour une bibliographie). Ces tests ont montré le réel intérêt pratique de ce type d'algorithme pour réduire sensiblement la variance (comme le *bagging*) mais aussi le biais de prédiction comparativement à d'autres approches. Cet algorithme est même considéré comme la meilleure méthode "*off-the-shelf*" c'est-à-dire ne nécessitant pas un long prétraitement des données ni un réglage fin de paramètres lors de la procédure d'apprentissage.

Le *boosting* adopte le même principe général que le *bagging* : construction d'une famille de modèles qui sont ensuite agrégés par une moyenne pondérée des estimations ou un vote. Il diffère nettement sur la façon de construire la famille qui est dans ce cas récurrente : chaque modèle est une version *adaptive* du précédent en donnant plus de poids, lors de l'estimation suivante, aux observations mal ajustées ou mal prédites. Intuitivement, cet algorithme concentre donc ses efforts sur les observations les plus difficiles à ajuster tandis que l'agrégation de l'ensemble des modèles permet d'échapper au sur-ajustement.

Les algorithmes de *boosting* proposés diffèrent par différentes caractéristiques :

- la façon de pondérer c'est-à-dire de renforcer l'importance des observations mal estimées lors de l'itération précédente,
- leur objectif selon le type de la variable à prédire Y : binaire, qualitative à k classes, réelles ;
- la fonction perte, qui peut être choisie plus ou moins robuste aux valeurs atypiques, pour mesurer l'erreur d'ajustement ;
- la façon d'agréger, ou plutôt pondérer, les modèles de base successifs.

La littérature sur le sujet présente donc de très nombreuses versions de cet algorithme et il est encore difficile de dire lesquelles sont les plus efficaces et si une telle diversité est bien nécessaire. Il serait fastidieux de vouloir expliciter toutes les versions, ce chapitre en propose un choix arbitraire.

3.2 Algorithme de base

Décrivons la version originale du *boosting* pour un problème de discrimination élémentaire à deux classes en notant δ la fonction de discrimination à valeurs dans $\{-1, 1\}$. Dans cette version, le modèle de base retourne l'identité d'une classe, il est encore nommé Adaboost discret. Il est facile de l'adapter à des modèles retournant une valeur réelle comme une probabilité d'appartenance à une classe.

ALGORITHME 7.3 : **AdaBoost** (adaptive boosting)

- Soit \mathbf{x}_0 à prévoir et
 - $\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ un échantillon
 - Initialiser les poids $\mathbf{w} = \{w_i = 1/n ; i = 1, \dots, n\}$.
 - Pour $m = 1$ à M Faire
 - Estimer δ_m sur l'échantillon pondéré par \mathbf{w} .
 - Calculer le taux d'erreur apparent :

$$\hat{\mathcal{E}}_p = \frac{\sum_{i=1}^n w_i \mathbf{1}\{\delta_m(\mathbf{x}_i) \neq y_i\}}{\sum_{i=1}^n w_i}$$
 - Calculer les logit : $c_m = \log((1 - \hat{\mathcal{E}}_p)/\hat{\mathcal{E}}_p)$.
 - Calculer les nouvelles pondérations : $w_i \leftarrow w_i \cdot \exp[c_m \mathbf{1}\{\delta_m(\mathbf{x}_i) \neq y_i\}] ; i = 1, \dots, n$.
 - Fin Pour
- Résultat du vote : $\hat{\phi}_M(\mathbf{x}_0) = \text{signe} \left[\sum_{m=1}^M c_m \delta_m(\mathbf{x}_0) \right]$.
-

Les poids de chaque observations sont initialisés à $1/n$ pour l'estimation du premier modèle puis évoluent à chaque itération donc pour chaque nouvelle estimation. L'importance d'une observation w_i est inchangée si elle est bien classée, elle croît sinon proportionnellement au défaut d'ajustement du modèle. L'agrégation finale des prévisions : $\sum_{m=1}^M c_m \delta_m(\mathbf{x}_0)$ est une combinaison pondérée par les qualités d'ajustement de chaque modèle. Sa valeur absolue appelée *marge* est proportionnelle à la confiance que l'on peut attribuer à son signe qui fournit le résultat de la prévision.

Ce type d'algorithme est largement utilisé avec un arbre (CART) comme modèle de base. De nombreuses applications montrent que si le "classifieur faible" est un arbre trivial à deux feuilles (*stump*), *AdaBoost* fait mieux qu'un arbre sophistiqué pour un volume de calcul comparable : autant de feuilles dans l'arbre que d'itérations dans *AdaBoost*. Hastie

et col. (2001) discutent la meilleure stratégie d'élagage applicable à chaque modèle de base. Ils le comparent avec le niveau d'interaction requis dans un modèle d'analyse de variance. Le cas $q = 2$ correspondant à la seule prise en compte des effets principaux. Empiriquement ils recommandent une valeur comprise entre 4 et 8.

3.3 Version aléatoire

À la suite de Freund et Schapire (1996), Breiman (1998) développe aussi, sous le nom d'*Arcing* (adaptively resample and combine), une version aléatoire, et en pratique très proche, du *boosting*. Elle s'adapte à des classifieurs pour lesquels il est difficile voire impossible d'intégrer une pondération des observations dans l'estimation. Ainsi plutôt que de jouer sur les pondérations, à chaque itération, un nouvel échantillon est tiré avec remise, comme pour le bootstrap, mais selon des probabilités inversement proportionnelles à la qualité d'ajustement de l'itération précédente. La présence des observations difficiles à ajuster est ainsi renforcée pour que le modèle y consacre plus d'attention. L'algorithme *adaboost* précédent est facile à adapter en ce sens en regardant celui développé ci-dessous pour la régression et qui adopte ce point de vue.

3.4 Pour la régression

Différentes adaptations du *boosting* ont été proposées pour le cas de la régression, c'est-à-dire lorsque la variable à prédire est quantitative. Voici l'algorithme de Drucker (1997) dans la présentation de Gey et Poggi (2002) qui en étudient les performances empiriques en relation avec CART. Freund et Schapire (1996) ont proposé *Adaboost.R* avec le même objectif tandis que le point de vue de Friedman (2002) est décrit plus loin dans l'algorithme 7.5.

ALGORITHME 7.4 : Boosting pour la régression

- Soit \mathbf{x}_0 à prévoir et
- $\mathbf{z} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ un échantillon
- Initialiser \mathbf{p} par la distribution uniforme $\mathbf{p} = \{p_i = 1/n ; i = 1, \dots, n\}$.
- Pour $m = 1$ à M Faire
 - Tirer avec remise dans \mathbf{z} un échantillon \mathbf{z}_m^* suivant \mathbf{p} .
 - Estimer $\hat{\phi}_m$ sur l'échantillon \mathbf{z}_m^* .
 - Calculer à partir de l'échantillon initial \mathbf{z} :

$$l_m(i) = Q\left(y_i, \hat{\phi}_m(\mathbf{x}_i)\right) \quad i = 1, \dots, n; \quad (Q : \text{fonction perte})$$

$$\widehat{\mathcal{E}}_m = \sum_{i=1}^n p_i l_m(i);$$

$$w_i = g(l_m(i))p_i. \quad (g \text{ continue non décroissante})$$

- Calculer les nouvelles probabilités : $p_i \leftarrow \frac{w_i}{\sum_{i=1}^n w_i}$.
- Fin Pour
- Calculer $\hat{\phi}(\mathbf{x}_0)$ moyenne ou médiane des prévisions $\hat{\phi}_m(\mathbf{x}_0)$ pondérées par des coefficients $\log(\frac{1}{\beta_m})$.

Précisions :

- Dans cet algorithme la fonction perte Q peut être exponentielle, quadratique ou, plus robuste, la valeur absolue. Le choix usuel de la fonction quadratique est retenu par Gey et Poggi (2002).
- Notons $L_m = \sup_{i=1,\dots,n} l_m(i)$ le maximum de l'erreur observée par le modèle $\hat{\phi}_m$ sur l'échantillon initial. La fonction g est définie par :

$$g(l_m(i)) = \beta_m^{1-l_m(i)/L_m} \quad (1)$$

$$\text{avec } \beta_m = \frac{\widehat{\mathcal{E}}_m}{L_m - \widehat{\mathcal{E}}_m}. \quad (2)$$

- Selon les auteurs, une condition supplémentaire est ajoutée à l'algorithme. Il est arrêté ou réinitialisé à des poids uniformes si l'erreur se dégrade trop : si $\widehat{\mathcal{E}}_m < 0.5L_m$.

L'algorithme génère M prédicteurs construits sur des échantillons bootstrap \mathbf{z}_m^* dont le tirage dépend de probabilités \mathbf{p} mises à jour à chaque itération. Cette mise à jour est fonction d'un paramètre β_m qui est un indicateur de la performance, sur l'échantillon \mathbf{z} , du m ème prédicteur estimé sur l'échantillon \mathbf{z}_m^* . La mise à jour des probabilités dépend donc à la fois de cet indicateur global β_m et de la qualité relative $l_m(i)/L_m$ de l'estimation du i ème individu. L'estimation finale est enfin obtenue à la suite d'une moyenne ou médiane des prévisions pondérées par la qualité respective de chacune de ces prévisions. Gey et Poggi (2002) conseille la médiane afin de s'affranchir de l'influence de prédicteurs très atypiques.

3.5 Modèle additif pas à pas

Hastie et col. (2001) expliquent le bon comportement du *boosting* dans le cas binaire en le présentant sous la forme d'une approximation de la fonction ϕ par un modèle additif construit pas à pas :

$$\hat{\phi}(\mathbf{x}) = \sum_{m=1}^M c_m \delta(\mathbf{x}; \gamma_m)$$

est cette combinaison où c_m est un paramètre, δ le classifieur (faible) de base fonction de \mathbf{x} et dépendant d'un paramètre γ_m . Si Q est une fonction perte, il s'agit, à chaque étape, de résoudre :

$$(c_m, \gamma_m) = \arg \min_{(c, \gamma)} \sum_{i=1}^n Q(y_i, \hat{\phi}_{m-1}(\mathbf{x}_i) + c\delta(\mathbf{x}_i; \gamma));$$

$\hat{\phi}_m(\mathbf{x}) = \hat{\phi}_{m-1}(\mathbf{x}) + c_m \delta(\mathbf{x}; \gamma_m)$ est alors une amélioration de l'ajustement précédent.

Dans le cas d'*adaboost* pour l'ajustement d'une fonction binaire, la fonction perte utilisée est $Q(y, \phi(\mathbf{x})) = \exp[-y\phi(\mathbf{x})]$. il s'agit donc de résoudre :

$$\begin{aligned} (c_m, \gamma_m) &= \arg \min_{(c, \gamma)} \sum_{i=1}^n \exp \left[-y_i \hat{\phi}_{m-1}(\mathbf{x}_i) + c\delta(\mathbf{x}_i; \gamma) \right]; \\ &= \arg \min_{(c, \gamma)} \sum_{i=1}^n w_i^m \exp [-cy_i \delta(\mathbf{x}_i; \gamma)] \\ \text{avec } w_i &= \exp[-y_i \hat{\phi}_{m-1}(\mathbf{x}_i)]; \end{aligned}$$

w_i ne dépendant ni de c ni de γ , il joue le rôle d'un poids fonction de la qualité de l'ajustement précédent. Quelques développements complémentaires montrent que la solution du problème de minimisation est obtenue en deux étapes : recherche du classifieur optimal puis optimisation du paramètre γ .

$$\begin{aligned}\gamma_m &= \arg \min_{\gamma} \sum_{i=1}^n \mathbf{1}\{y_i \neq \delta(\mathbf{x}_i; \gamma)\}, \\ c_m &= \frac{1}{2} \log \frac{1 - \hat{\mathcal{E}}_p}{\mathcal{E}_p}\end{aligned}$$

avec $\hat{\mathcal{E}}_p$ erreur apparente de prédiction tandis que les w_i sont mis à jour avec :

$$w_i^{(m)} = w_i^{(m-1)} \exp[-c_m].$$

On montre ainsi qu'*adaboost* approche ϕ pas à pas par un modèle additif en utilisant une fonction perte exponentielle.

D'autres fonctions perte sont envisageables pour, en particulier, un algorithme plus robuste face à un échantillon d'apprentissage présentant des erreurs de classement dans le cas de la discrimination ou encore des valeurs atypiques (*outliers*) dans le cas de la régression. Hastie et col. (2001) comparent les intérêts respectifs de plusieurs fonctions pertes. Celles jugées robustes (entropie en discrimination, valeur absolue en régression) conduisent à des algorithmes plus compliqués à mettre en œuvre.

3.6 Régression et boosting

Dans le même esprit d'approximation adaptative, Friedman (2002) propose sous l'acronyme MART (*multiple additive regression trees*) un algorithme basé sur des arbres de régression pour traiter le cas quantitatif en supposant la fonction perte seulement différentiable. Le principe de base est le même que pour *Adaboost*, construire une séquence de modèles de sorte que chaque étape, chaque modèle ajouté à la combinaison, apparaisse comme un pas vers une meilleure solution. Ce pas est franchi dans la direction du gradient, approché par un arbre de régression, de la fonction perte.

ALGORITHME 7.5 : **MART** (Multiple additive regression trees)

- Soit \mathbf{x}_0 à prévoir
- Initialiser $\hat{\phi}_0 = \arg \min_{\gamma} \sum_{i=1}^n Q(y_i, \gamma)$
- Pour $m = 1$ à M Faire
 - Calculer $r_m = - \left[\frac{\delta Q(y_i, \phi(\mathbf{x}_i))}{\delta \phi(\mathbf{x}_i)} \right]_{\phi=\phi_{m-1}}$,
 - Ajuster un arbre de régression aux r_m donnant les feuilles ou régions terminales $R_{jm}; j = 1, \dots, J_m$.
 - Pour $m = 1$ à M Faire
 - Calculer $\gamma_{jm} = \arg \min_{\gamma} \sum_{\mathbf{x}_i \in R_{jm}} Q(y_i, \hat{\phi}_{m-1} + \gamma)$.
 - Fin Pour
 - Mise à jour : $\hat{\phi}_m(\mathbf{x}) = \hat{\phi}_{m-1}(\mathbf{x}) + \sum_{j=1}^{J_m} \gamma_{jm} \mathbf{1}\{\mathbf{x} \in R_{jm}\}$.
- Fin Pour
- Résultat : $\hat{\phi}_M(\mathbf{x}_0)$.

L'algorithme est initialisé par un terme constant c'est-à-dire encore un arbre à une feuille. Les expressions du gradient reviennent simplement à calculer les résidus r_{mj} du modèle à l'étape précédente. Les termes correctifs γ_{jm} sont ensuite optimisés pour chacune des régions R_{jm} définies par l'arbre de régression ajustant les résidus. Un algorithme de discrimination est similaire calculant autant de probabilités que de classes à prévoir.

3.7 Compléments

De nombreuses adaptations ont été proposées à partir de l'algorithme initial. Elles font intervenir différentes fonctions pertes offrant des propriétés de robustesse ou adaptées à une variable cible Y quantitative ou qualitative à plusieurs classes : *Adaboost* M1, M2, MH ou encore MR. Schapire (2002) liste une bibliographie détaillée.

Sur-ajustement

Dans le dernier algorithme, le nombre d'itérations peut être contrôlé par un échantillon de validation. Comme pour d'autres méthodes (perceptron), il suffit d'arrêter la procédure lorsque l'erreur estimée sur cet échantillon arrive à se dégrader. Une autre possibilité consiste à ajouter un coefficient de rétrécissement (*shrinkage* comme en régression ridge). Compris entre 0 et 1, celui-ci pénalise l'ajout d'un nouveau modèle dans l'agrégation. Il joue le rôle d'un taux d'apprentissage du perceptron) et, si sa valeur est petite ($< 0,1$) cela conduit à accroître le nombre d'arbres mais entraîne des améliorations sensibles de la qualité de prédiction.

Interprétation

L'interprétabilité des arbres de décision sont une des raisons de leur succès. Leur lecture ne nécessite pas de compétences particulières en statistique. Cette propriété est évidemment perdue par l'agrégation d'arbres ou de tout autre modèle. Néanmoins, surtout si le nombre de variables est très grand, il est important d'avoir une indication de l'importance relative des variables entrant dans la modélisation.

Un critère est calculé pour chaque variable j à partir des valeurs $D_j^2(l, m)$, calculées pour chaque nœud l de chaque arbre m . Cette quantité est la décroissance optimale de déviance produite par la segmentation associée à ce nœud par le choix de la variable j . Ces valeurs sont sommées par arbre sur l'ensemble des nœuds puis moyennées sur l'ensemble des arbres. Une normalisation fixe à 100 la plus grande valeur correspondant à la variable la plus influente.

Instabilité

Tous les auteurs ont remarqué la grande instabilité des modèles construits à base d'arbres : une légère modification des données est susceptible d'engendrer de grandes modifications dans les paramètres (les seuils et feuilles) du modèle. C'est justement cette propriété qui rend cette technique très appropriée à une amélioration par agrégation. Breiman (1998), pour les arbres de classification, puis Gey et Poggi (2002), pour les arbres

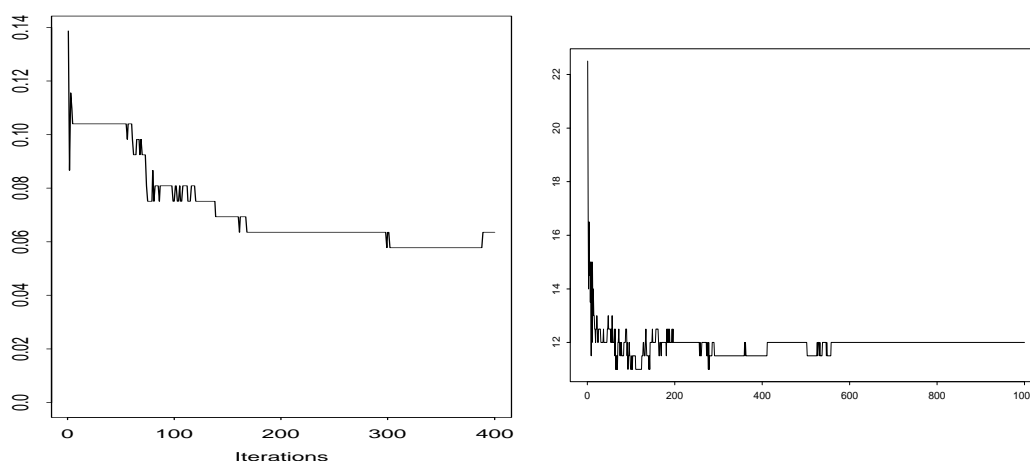


FIG. 7.1 – Algorithmes AdaBoost et Random forests. Évolution, pour un échantillon test, du taux de mal classés en fonction du nombre d'arbres intervenant dans la combinaison de modèles.

de régression, détaillent et quantifient en pratique l'influence de cette instabilité ainsi que celle de l'apport potentiel du *boosting* par rapport au *bagging*.

Propriétés

Les justifications théoriques des bons résultats du *boosting* et principalement la résistance au sur-ajustement sont encore l'objet de travaux intenses suivant différentes pistes. La difficulté vient de ce que l'application de ce type d'algorithme sur une méthode donnée, fait généralement mieux que l'asymptotique (en faisant croître la taille de l'échantillon) pour cette même méthode. Les approches usuelles de la statistique asymptotique sont mises en défaut et les bornes obtenues pour majorer les erreurs d'estimations ou de prédiction sont trop grossières pour rendre compte de l'efficacité effective de la méthode. On montre ainsi, empiriquement, que l'erreur de prédiction ou de généralisation peut continuer à décroître longtemps après que l'erreur d'ajustement se soit annulée. Parmi les pistes explorées, une approche "stochastique" considère que, même déterministe, l'algorithme simule une dynamique markovienne (Blanchard, 2001). Une deuxième, rappelée ci-dessus, présente le *boosting* comme une procédure d'optimisation globale par une méthode de gradient (Friedman, 2001). D'autres enfin (par exemple Lugosi et Vayatis, 2001), plus probantes, utilisent des inégalités de Vapnik pour montrer que, sous des hypothèses raisonnables et vérifiées dans les cas usuels : convexité et régularité de la fonction perte (exponentielle), arbres binaires, la probabilité d'erreur du *boosting* converge avec la taille n de l'échantillon vers celle du classifieur bayésien c'est-à-dire celui, optimal, obtenu en supposant connue la loi conjointe de X et Y .

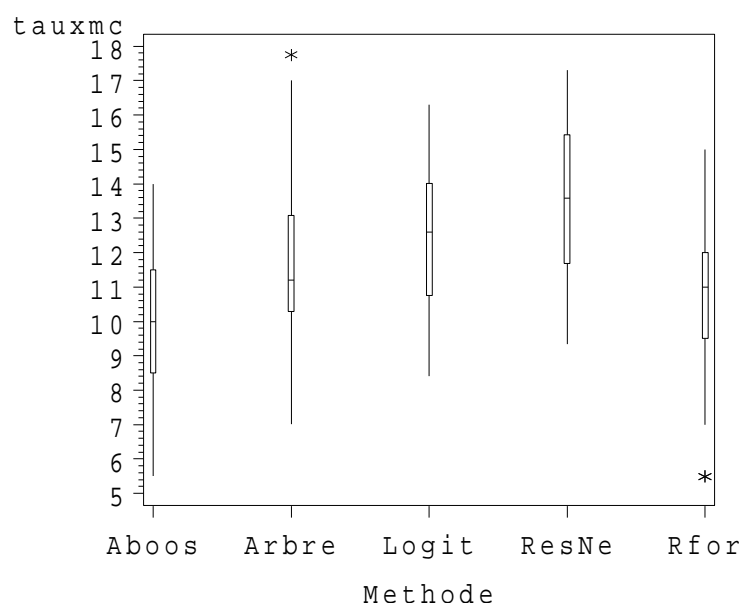


FIG. 7.2 – Diagrammes boîtes des taux d’erreurs observés sur 30 échantillons tests et pour chaque méthode.

4 Application

4.1 Logiciels

L’algorithme de *boosting* (Freund et Schapire, 1996), ou plutôt la version de Friedman et col. (2000) a été utilisée. Il en fournissait le programme interfacé avec Splus à l’adresse www-stat.stanford.edu/~jhf/MART.html mais ce dernier est maintenant commercialisé par la société *Salford System*. En revanche, R. Schapire diffuse le logiciel *Boost texter* sur sa page : www.research.att.com/~schapire pour des utilisations non commerciales. D’autres versions du *boosting* seront vraisemblablement rendues disponibles dans l’environnement de R (clone de Splus sous licence GNU).

Les forêts aléatoires (Breiman, 2001), sont estimées par un programme interfacé avec R qui se trouve dans la page : www.stat.Berkeley.edu/users/breiman/ ou avec la distribution de R : www.r-project.org.

4.2 Résultats comparatifs

Ces programmes ont été utilisés pour constituer des comités d’arbres de décision prévoyant la possession de la carte Visa Premier. Trente échantillons tests ont successivement été tirés afin d’observer les distributions des taux de mauvais classement, distributions qui ont été comparées à celles obtenues par les méthodes classiques (arbre de décision, régression logistique et réseaux de neurones).

La figure 7.1 montre les évolutions du taux de mal classés sur l’échantillon d’apprentissage en fonction du nombre d’arbres estimés pour un exemple de tirage. Malgré la complexité des combinaisons de modèles finalement obtenues, le taux atteint une limite, il n’y a pas sur-apprentissage. Ces algorithmes fournissent des résultats qui, en moyenne, se

TAB. 7.1 – Moyennes des taux d’erreurs de classement calculés sur 30 échantillons test pour chaque modèle de prédiction

Méthode	Adaboost	Arbre	Régression	Perceptron	Forêt
Moyenne	9.7	11.8	12.5	13.4	10.6
Écart-type	2.0	2.3	2.0	2.3	2.2

montrent sensiblement plus performants (cf. figure 7.2 et tableau 7.1) sur un échantillon test. Les écarts-types dépendant de la taille de l’échantillon test y sont relativement stables. Les moyennes montrent, sur cet exemple, que le *boosting* prédit un peu mieux que les forêts aléatoires. Cela est cohérent avec les nombreuses études publiées.

Bien sûr, ce qui est gagné en prédictibilité est perdu en interprétabilité par rapport à un modèle classique. Néanmoins le gain réalisé est souvent étonnant. L’une des avancées encore en gestation concernant ces algorithmes, et plus particulièrement les forêts aléatoires, est la prise en compte des problèmes posés par les données hautement multidimensionnelles tels qu’ils se posent par exemple avec l’analyse des biopuces en génomique.

Bibliography

- Agresti, A. (1990). *Categorical data analysis*. Wiley.
- Antoniadis, A., J. Berruyer, and R. Carmona (1992). *Régression non linéaire et applications*. Economica.
- Baccini, A. and P. Besse (2000). Data mining : 1. exploration statistique. www.ups-tlse.fr/Besse/enseignement.html.
- Besse, P. and H. Cardot (2003). Modélisation statistique de données fonctionnelles. In G. Govaert (Ed.), *Analyse des données*, pp. à paraître. Hermes.
- Besse, P., C. Le Gall, N. Raimbault, and S. Sarpy (2001). Statistique et data mining. *Journal de la Société Française de Statistique* 142, 5–36.
- Blanchard, G. (2001). Generalization error bounds for aggregate classifiers. In *Proceedings of the MSRI international conference on nonparametric estimation and classification*, pp. .
- Breiman, L. (1996). Bagging predictors. *Machine Learning* 26(2), 123–140.
- Breiman, L. (1998). Arcing classifiers. *Annals of Statistics* 26, 801–849.
- Breiman, L. (2001). Random forests random features. *Machine Learning à paraître*, .
- Breiman, L., J. Friedman, R. Olshen, and C. Stone (1984). *Classification and regression trees*. Wadsworth & Brooks.
- Drucker, H. (1997). Improving regressors using boosting techniques. In M. Kaufmann (Ed.), *Proceedings of the 14th International Conference on Machine Learning*, pp. 107–115.
- Efron, B. (1982). *The Jackknife, the Bootstrap and other Resampling Methods*. SIAM.
- Efron, B. and R. Tibshirani (1993). *An introduction to the bootstrap*. Chapman and Hall.
- Freund, Y. and R. Schapire (1996). Experiments with a new boosting algorithm. In *Machine Learning : proceedings of the Thirteenth International Conference*, pp. 148–156. Morgan Kaufman. San Francisco.
- Freund, Y. and R. Schapire (1997). Experiments with a new boosting algorithm. *Journal of Computer and System Sciences* 55, 119–139.
- Friedman, J. H. (2001). Greedy function approximation : a gradient boosting machine. *Annals of Statistics* 29, 1189–1232.
- Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics and Data Analysis* 38, .
- Friedman, J. H., H. Hastie, and R. Tibshirani (2000). Additive logistic regression : a statistical view of boosting. *The Annals of Statistics* 28, 337–407.

- Gey, S. and J.-M. Poggi (2002). Boosting and instabillity for regression trees. Technical Report 36, Université de Paris Sud, Mathématiques.
- Ghattas, B. (2000). Agrégation d'arbres de classification. *Revue de Statistique Appliquée* 48(2), 85–98.
- Hastie, T., R. Tibshirani, and J. Friedman (2001). *The elements of statistical learning : data mining, inference, and prediction*. Springer.
- Haykin, T. (1994). *Neural network, a comprehensive foundation*. Prentice-Hall.
- Jobson, J. (1991). *Applied Multivariate Data Analysis*, Volume I : Regression and experimental design. Springer-Verlag.
- Lugosi, G. and N. Vayatis (2001). On the bayes-risk consistency of boosting methods. *Preprint* , .
- McCullagh, P. and J. Nelder (1983). *Generalized Linear Models*. Chapman & Hall.
- Quinlan, J. (1993). *C4.5 – Programs for machine learning*. Morgan Kaufmann.
- Ripley, B. (1996). *Pattern recognition and neural networks*. Cambridge University Press.
- SAS (1989). *SAS/STAT User's Guide* (fourth ed.), Volume 2. Sas Institute Inc. version 6.
- SAS (1995). *SAS/INSIGHT User's Guide* (Third ed.). Sas Institute Inc. version 6.
- Schapire, R. (1990). The strength of weak learnability. *Machine Learning* 5, 197–227.
- Schapire, R. (2002). The boosting approach to machine learning. an overview. In *MSRI workshop on non linear estimation and classification*, pp. .
- SEM (2001). *SAS/ Enterprise Miner User's Guide*. Sas Institute Inc. version 8.
- Thiria, S., Y. Lechevallier, O. Gascuel, and S. Canu (1997). *Statistique et méthodes neuronales*. Dunod.
- Vapnik, V. (1999). *Statistical learning theory*. Wiley Inter science.

Annexes

Annexe A

Introduction au modèle linéaire général

L'objet de ce chapitre est d'introduire le cadre théorique global permettant de regrouper tous les modèles (linéaire gaussien, logit, log-linéaire) qui visent à exprimer l'espérance d'une variable réponse Y en fonction d'une combinaison linéaire des variables explicatives. Le *modèle linéaire général* développé initialement en 1972 par Nelder et Wedderburn et dont on trouvera des exposés détaillés dans Nelder et Mc Cullagh (1983), Agresti (1990) ou Antoniadis et col. (1992), n'est ici qu'esquissé afin de définir les concepts communs à ces modèles : famille exponentielle, estimation par maximum de vraisemblance, tests, diagnostics, résidus. Il est mis en œuvre dans plusieurs logiciels dont GLIM, `glm` de Splus, `genmod` et `insight` de SAS.

1 Composantes des modèles

Les modèles catalogués dans la classe des modèles linéaires généralisés sont caractérisés par trois composantes.

1.1 Distribution

La *composante aléatoire* identifie la distribution de probabilités de la variable à expliquer. On suppose que l'échantillon statistique est constitué de n variables aléatoires $\{Y_i; i = 1, \dots, n\}$ indépendantes admettant des distributions issues d'une *structure exponentielle*. Cela signifie que les lois de ces variables sont dominées par une même mesure dite de référence et que la famille de leurs densités par rapport à cette mesure se met sous la forme :

$$f(y_i; \theta_i, \phi) = \exp \left\{ \frac{y_i \theta_i - v(\theta_i)}{u(\phi)} + w(y_i, \phi) \right\}. \quad (1)$$

Cette formulation inclut la plupart des lois usuelles comportant un ou deux paramètres : gaussienne, gaussienne inverse, gamma, Poisson, binomiale. . . . Le paramètre θ_i est appelé *paramètre naturel* de la famille exponentielle.

Attention, la mesure de référence change d'une structure exponentielle à l'autre, la mesure de Lebesgues pour une loi continue, une mesure discrète combinaison de masses

de Dirac pour une loi discrète. Consulter Antoniadis et col. (1992) pour une présentation générale des structures exponentielles et des propriétés asymptotiques des estimateurs de leurs paramètres.

Pour certaines lois, la fonction u est de la forme :

$$u(\phi) = \frac{\phi}{\omega_i}$$

où les poids ω_i sont les poids connus des observations, fixés ici à 1 pour simplifier ; ϕ est appelé alors *paramètre de dispersion*, c'est un paramètre de nuisance intervenant, par exemple lorsque les variances des lois gaussiennes sont inconnues, mais égal à 1 pour les lois à un paramètre (Poisson, binomiale). L'expression de la structure exponentielle (1) se met alors sous la *forme canonique* en posant :

$$\begin{aligned} Q(\theta) &= \frac{\theta}{\phi}, \\ a(\theta) &= \exp \left\{ -\frac{v(\theta)}{\phi} \right\}, \\ b(y) &= \exp \{ w(y, \phi) \}, \end{aligned}$$

on obtient

$$f(y_i, \theta_i) = a(\theta_i) b(y_i) \exp \{ y_i Q(\theta_i) \}. \quad (2)$$

1.2 Prédicteur linéaire

Les observations planifiées des variables explicatives sont organisées dans la matrice \mathbf{X} de planification d'expérience (design matrix). Soit $\boldsymbol{\beta}$ un vecteur de p paramètres, le prédicteur linéaire, *composante déterministe* du modèle, est le vecteur à n composantes :

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}.$$

1.3 Lien

La troisième composante exprime une *relation fonctionnelle* entre la composante aléatoire et le prédicteur linéaire. Soit $\{\mu_i = E(Y_i); i = 1, \dots, n\}$, on pose

$$\eta_i = g(\mu_i) \quad i = 1, \dots, n$$

où g , appelée *fonction lien*, est supposée monotone et différentiable. Ceci revient donc à écrire un modèle dans lequel une *fonction de la moyenne* appartient au sous-espace engendré par les variables explicatives :

$$g(\mu_i) = \mathbf{x}_i' \boldsymbol{\beta} \quad i = 1, \dots, n.$$

La fonction lien qui associe la moyenne μ_i au paramètre naturel est appelée *fonction lien canonique*. Dans ce cas,

$$g(\mu_i) = \theta_i = \mathbf{x}_i' \boldsymbol{\beta}.$$

1.4 Exemples

Loi gaussienne

Dans le cas d'un échantillon gaussien, les densités d'une famille de lois $\mathcal{N}(\mu_i, \sigma^2)$ s'écrit :

$$\begin{aligned} f(y_i, \mu_i) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(y_i - \mu_i)^2}{2\sigma^2} \right\} \\ &= \exp \left\{ -\frac{1}{2} \frac{\mu_i^2}{\sigma^2} \right\} \exp \left\{ -\frac{1}{2} \frac{y_i^2}{\sigma^2} - \frac{1}{2} \ln(2\pi\sigma^2) \right\} \exp \left\{ y_i \frac{\mu_i}{\sigma^2} \right\} \end{aligned}$$

En posant

$$\begin{aligned} Q(\theta_i) &= \frac{\theta_i}{\phi} = \frac{\mu_i}{\sigma^2} \\ a(\theta_i) &= \exp \left\{ -\frac{1}{2} \frac{\mu_i^2}{\sigma^2} \right\} \\ b(y_i) &= \exp \left\{ -\frac{1}{2} \frac{y_i^2}{\sigma^2} - \frac{1}{2} \ln(2\pi\sigma^2) \right\}. \end{aligned}$$

la famille gaussienne se met sous la forme canonique (2) qui en fait une famille exponentielle de paramètre de dispersion $\phi = \sigma^2$ et de paramètre naturel

$$\theta_i = E(Y_i) = \mu_i$$

et donc de fonction lien canonique, la fonction *identité*.

Loi de Bernoulli

Considérons n variables aléatoires binaires indépendantes Z_i de probabilité de succès π_i et donc d'espérance $E(Z_i) = \pi_i$. Les fonctions de densité de ces variables sont éléments de la famille :

$$f(z_i, \pi_i) = \pi_i^{z_i} (1 - \pi_i)^{1-z_i} = (1 - \pi_i) \exp \left\{ z_i \ln \frac{\pi_i}{1 - \pi_i} \right\},$$

qui est la forme canonique d'une structure exponentielle de paramètre naturel

$$\theta_i = \ln \frac{\pi_i}{1 - \pi_i}.$$

Cette relation définit la fonction *logit* pour fonction lien canonique associée à ce modèle. La loi binomiale conduit à des résultats identiques en considérant les sommes de n_i (n_i connus) variables de Bernoulli.

Loi de Poisson

On considère n variables indépendantes Y_i de loi de Poisson de paramètre $\mu_i = E(Y_i)$. Les Y_i sont par exemple les effectifs d'une table de contingence. Ces variables admettent pour densités :

$$f(y_i, \mu_i) = \frac{\mu_i^{y_i} e^{-\mu_i}}{y_i!} = \exp \{-\mu_i\} \frac{1}{y_i!} \exp \{y_i \ln \mu_i\}$$

qui sont issues d'une structure exponentielle et, mises sous la forme canonique, de paramètre naturel

$$\theta_i = \ln \mu_i$$

définissant comme fonction lien canonique le *logarithme* pour ce modèle.

2 Estimation

L'estimation des paramètres β_j est calculée en maximisant la log-vraisemblance du modèle linéaire généralisé. Celle-ci s'exprime pour toute famille de distributions mise sous la forme (1) d'une structure exponentielle.

2.1 Expression des moments

Notons $\ell(\theta_i, \phi; y_i) = \ln f(y_i; \theta_i, \phi)$ la contribution de la i ème observation à la log-vraisemblance.

$$\ell(\theta_i, \phi; y_i) = [y_i \theta_i - v(\theta_i)]/u(\phi) + w(y_i, \phi).$$

L'étude du maximum de la log-vraisemblance nécessite la connaissance des dérivées :

$$\begin{aligned} \frac{\partial \ell}{\partial \theta_i} &= [y_i - v'(\theta_i)]/u(\phi) \\ \frac{\partial^2 \ell}{\partial \theta_i^2} &= -v''(\theta_i)/u(\phi). \end{aligned}$$

Pour des lois issues de structures exponentielles, les conditions de régularité vérifiées permettent d'écrire :

$$E\left(\frac{\partial \ell}{\partial \theta}\right) = 0 \quad \text{et} \quad -E\left(\frac{\partial^2 \ell}{\partial \theta^2}\right) = E\left(\frac{\partial \ell}{\partial \theta}\right)^2.$$

Alors,

$$E(Y_i) = \mu_i = v'(\theta_i)$$

et comme

$$E\{v''(\theta_i)/u(\phi)\} = E\{[Y_i - v'(\theta_i)]/u(\phi)\}^2 = \text{Var}(Y_i)/u^2(\phi)$$

il vient donc :

$$\text{Var}(Y_i) = v''(\theta_i)u(\phi) ;$$

justifiant ainsi l'appellation de *paramètre de dispersion* pour ϕ lorsque u est la fonction identité.

2.2 Équations de vraisemblance

Considérons p variables explicatives dont les observations sont rangées dans la matrice de plan d'expérience \mathbf{X} , $\boldsymbol{\beta}$ un vecteur de p paramètres et le prédicteur linéaire à n composantes

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}.$$

La fonction lien g est supposée monotone différentiable telle que : $\eta_i = g(\mu_i)$;

c'est la fonction lien canonique si : $g(\mu_i) = \theta_i$.

Pour n observations supposées indépendantes et en tenant compte que $\boldsymbol{\theta}$ dépend de $\boldsymbol{\beta}$, la log-vraisemblance s'écrit :

$$\mathcal{L}(\boldsymbol{\beta}) = \sum_{i=1}^n \ln f(y_i; \boldsymbol{\theta}_i, \phi) = \sum_{i=1}^n \ell(\boldsymbol{\theta}_i, \phi; y_i).$$

Calculons

$$\frac{\partial \ell_i}{\partial \beta_j} = \frac{\partial \ell_i}{\partial \theta_i} \frac{\partial \theta_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \eta_i} \frac{\partial \eta_i}{\partial \beta_j}.$$

Comme

$$\begin{aligned} \frac{\partial \ell_i}{\partial \theta_i} &= [y_i - v'(\theta_i)]/u(\phi) = (y_i - \mu_i)/u(\phi), \\ \frac{\partial \mu_i}{\partial \theta_i} &= v''(\theta_i) = \text{Var}(Y_i)/u(\phi), \\ \frac{\partial \eta_i}{\partial \beta_j} &= x_{ij} \quad \text{car} \quad \eta_i = \mathbf{x}_i' \boldsymbol{\beta}, \\ \frac{\partial \mu_i}{\partial \eta_i} &\quad \text{dépend de la fonction lien} \quad \eta_i = g(\mu_i), \end{aligned}$$

Les équations de la vraisemblance sont :

$$\sum_{i=1}^n \frac{(y_i - \mu_i)x_{ij}}{\text{Var}(Y_i)} \frac{\partial \mu_i}{\partial \eta_i} = 0 \quad j = 1, \dots, p.$$

Ce sont des équations non-linéaires en $\boldsymbol{\beta}$ dont la résolution requiert des méthodes itératives dans lesquelles interviennent le Hessien (pour Newton-Raphson) ou la *matrice d'information* (pour les Scores de Fisher). La matrice d'information est la matrice

$$\mathfrak{S} = \mathbf{X}' \mathbf{W} \mathbf{X}$$

de terme général

$$[\mathfrak{S}]_{jk} = E \frac{\partial^2 \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_k} = - \sum_{i=1}^n \frac{x_{ij} x_{ik}}{\text{Var}(Y_i)} \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2$$

et où \mathbf{W} est la matrice diagonale de “pondération” :

$$[\mathbf{W}]_{ii} = \frac{1}{\text{Var}(Y_i)} \left(\frac{\partial \mu_i}{\partial \eta_i} \right)^2.$$

2.3 Fonction lien canonique

Dans le cas particulier où la fonction lien du modèle linéaire généralisé utilisée est la fonction lien canonique associée à la structure exponentielle alors plusieurs simplifications interviennent :

$$\begin{aligned} \eta_i &= \theta_i = \mathbf{x}_i' \boldsymbol{\beta}, \\ \frac{\partial \mu_i}{\partial \eta_i} &= \frac{\partial \mu_i}{\partial \theta_i} = \frac{\partial v'(\theta_i)}{\partial \theta_i} = v''(\theta_i). \end{aligned}$$

Ainsi,

$$\frac{\partial \ell_i}{\partial \beta_j} = \frac{(y_i - \mu_i)}{\text{Var}(Y_i)} v''(\theta_i) x_{ij} = \frac{(y_i - \mu_i)}{u(\phi)} x_{ij}.$$

De plus, comme les termes $\frac{\partial^2 \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_k}$ ne dépendent plus de y_i , on montre que le Hessien est égal à la matrice d'information et donc les méthodes de résolution du score de Fisher et de Newton-Raphson coïncident.

Si, de plus, $u(\phi)$ est constante pour les observations, les équations de vraisemblance deviennent :

$$\mathbf{X}'\mathbf{y} = \mathbf{X}'\boldsymbol{\mu}.$$

Ainsi, dans le cas gaussien, le modèle s'écrivant $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$ avec la fonction de lien canonique identité, on retrouve la solution :

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

qui coïncide avec celle obtenue par minimisation des moindres carrés.

3 Qualité d'ajustement

Il s'agit d'évaluer la qualité d'ajustement du modèle sur la base des différences entre observations et estimations. Plusieurs critères sont proposés.

3.1 Déviance

Le modèle estimé est comparé avec le modèle dit *saturé*, c'est-à-dire le modèle possédant autant de paramètres que d'observations et estimant donc exactement les données. Cette comparaison est basée sur l'expression de la *déviance* D des log-vraisemblances \mathcal{L} et \mathcal{L}_{sat} :

$$D = -2(\mathcal{L} - \mathcal{L}_{\text{sat}})$$

qui est le logarithme du carré du rapport des vraisemblances. Ce rapport remplace ou "généralise" l'usage des sommes de carrés propres au cas gaussien et donc à l'estimation par moindres carrés.

On montre qu'asymptotiquement, D suit une loi du χ^2 à $n - p$ degrés de liberté ce qui permet de construire un test de rejet ou d'acceptation du modèle selon que la déviance est jugée significativement ou non importante.

Attention, l'approximation de la loi du χ^2 peut être douteuse. De plus, dans le cas de données non groupées (modèle binomial), le cadre asymptotique n'est plus adapté car le nombre de paramètres estimés tend également vers l'infini avec n et il ne faut plus se fier à ce test.

3.2 Test de Pearson

Un test du χ^2 est également utilisé pour comparer les valeurs observées y_i à leur prévision par le modèle. La statistique du test est définie par

$$X^2 = \sum_{i=1}^I \frac{(y_i - \hat{\mu}_i)^2}{\widehat{\text{Var}}(\hat{\mu}_i)}$$

(μ_i est remplacé par $n_i\pi_i$ dans le cas binomial) et on montre qu'elle admet asymptotiquement la même loi que la déviance.

En pratique ces deux approches conduisent à des résultats peu différents et, dans le cas contraire, c'est une indication de mauvaise approximation de la loi asymptotique. Sachant que l'espérance d'une loi du χ^2 est son nombre de degrés de liberté et, connaissant les aspects approximatifs des tests construits, l'usage est souvent de comparer les statistiques avec le nombre de degrés de liberté. le modèle peut être jugé satisfaisant pour un rapport D/ddl plus petit que 1.

4 Tests

Deux critères sont habituellement proposés pour aider au choix de modèle.

4.1 Rapport de vraisemblance

Comme dans le cas de la régression multiple où un test permet de comparer un modèle avec un modèle réduit, le rapport de vraisemblance ou la différence de déviance est une évaluation de l'apport des variables explicatives supplémentaires dans l'ajustement du modèle. La différence des déviances entre deux modèles *emboîtés* respectivement à q_1 et q_2 ($q_2 > q_1$) variables explicatives

$$\begin{aligned} D_2 - D_1 &= 2(\mathcal{L}_1 - \mathcal{L}_{\text{sat}}) - 2(\mathcal{L}_2 - \mathcal{L}_{\text{sat}}) \\ &= 2(\mathcal{L}_1 - \mathcal{L}_2) \end{aligned}$$

suit approximativement une loi du χ^2 à $(q_2 - q_1)$ degrés de liberté pour les lois à 1 paramètre (binomial, Poisson) et une loi de Fisher pour les lois à deux paramètres (gaussienne). Ceci permet donc de tester la significativité de la diminution de la déviance par l'ajout de variables explicatives ou la prise en compte d'interactions.

4.2 Test de Wald

Ce test est basé sur la forme quadratique faisant intervenir la matrice de covariance des paramètres, l'inverse de la matrice d'information observée $(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}$. Cette matrice est calculée à partir du Hessien approché par l'algorithme de maximisation. Elle généralise la matrice $(\mathbf{X}'\mathbf{X})^{-1}$ utilisée dans le cas du modèle linéaire gaussien en faisant intervenir une matrice \mathbf{W} de pondération. Ainsi, test de Wald et test de Fisher sont équivalents dans le cas particulier du modèle gaussien.

Si la matrice \mathbf{K} , dite *contraste*, définit l'ensemble H_0 des hypothèses à tester sur les paramètres :

$$\mathbf{K}'\boldsymbol{\beta} = 0,$$

on montre que la statistique

$$(\mathbf{K}'\mathbf{b})'(\mathbf{K}'(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{K})^{-1}\mathbf{K}'\mathbf{b}$$

suit asymptotiquement une loi du χ^2 .

Attention, le test de Wald, approximatif, peut ne pas être précis si le nombre d'observations est faible.

5 Diagnostics

De nombreux indicateurs, comme dans le cas de la régression linéaire multiple, sont proposés afin d'évaluer la qualité ou la robustesse des modèles estimés. Ils concernent la détection des valeurs influentes et l'étude graphique des résidus. La définition de ces derniers pose quelques difficultés.

5.1 Effet levier

On construit la matrice de projection (hat matrix)

$$\mathbf{H} = \mathbf{W}^{1/2} \mathbf{X} (\mathbf{X}' \mathbf{W} \mathbf{X})^{-1} \mathbf{X}' \mathbf{W}^{1/2},$$

relative au produit scalaire de matrice \mathbf{W} , sur le sous-espace engendré par les variables explicatives. Les termes diagonaux de cette matrice supérieurs à $(3p/n)$ indiquent des valeurs potentiellement influentes. Le graphe représentant les points d'ordonnées h_{ii} et d'abscisses le numéro de l'observation les visualise.

5.2 Résidus

Avec des erreurs centrées, additives, c'est-à-dire dans le cas du modèle gaussien utilisant la fonction lien identité, il est naturel de définir des résidus par :

$$\varepsilon_i = y_i - E(y_i) = y_i - \mu_i.$$

comme dans le cas du modèle linéaire. Ce cadre est ici inadapté au cas général et différents substituts sont proposés. Chacun possède par ailleurs une version *standardisée* et une version *studentisée*.

Pearson

Les résidus obtenus en comparant valeurs observées y_i et valeurs prédites \hat{y}_i sont pondérés par leur précision estimée par l'écart-type : s_i de \hat{y}_i . Ceci définit les résidus de Pearson :

$$r_{Pi} = \frac{y_i - \hat{y}_i}{s_i}$$

dont la somme des carrés conduit à la statistique du même nom. Ces résidus mesurent donc la contribution de chaque observation à la significativité du test découlant de cette statistique. Par analogie au modèle linéaire, on vérifie que ce sont également les résidus de la projection par la matrice \mathbf{H} .

Ces résidus ne sont pas de variance unité et sont donc difficiles à interpréter. Une estimation de leurs écarts-types conduit à la définition des résidus de Pearson standardisés :

$$r_{Psi} = \frac{y_i - \hat{y}_i}{s_i \sqrt{h_{ii}}}$$

faisant intervenir le terme diagonal de la matrice \mathbf{H} .

De plus, prenant en compte que les estimations des écarts-types s_i dépendent de la i ème observation et sont donc biaisés, des résidus studentisés sont obtenus en approchant au premier ordre le paramètre de dispersion $s_{(i)}$ calculé sans la i ème observation :

$$r_{Pti} = \frac{y_i - \hat{y}_i}{s_{(i)}\sqrt{h_{ii}}}.$$

Déviance

Ces résidus mesurent la contribution de chaque observation à la déviance du modèle par rapport au modèle saturé. Des versions standardisées et studentisées en sont définies comme pour ceux de Pearson.

Anscombe

Les lois des résidus précédents sont inconnues et même dissymétriques. Anscombe a donc proposé de faire opérer une transformation préalable afin de construire des résidus suivant une loi normale :

$$r_{Ai} = \frac{t(y_i) - t(\hat{y}_i)}{t'(y_i)s_i}.$$

L'explicitation de la fonction t dans le cadre du modèle linéaire généralisé est relativement complexe mais le calcul en est fourni par les logiciels. Comme précédemment, des versions standardisées et studentisées sont également calculées.

Un graphe utilisant ces résidus en ordonnées et les numéros d'observation en abscisses permet d'identifier les observations les moins bien ajustées par le modèle.

5.3 Mesure d'influence

De nombreux indicateurs sont proposés afin d'évaluer l'influence d'une observation sur l'estimation d'un paramètre, sur les prédictions ou encore sur la variance des estimateurs. Le plus utilisé, la distance de Cook, mesure globalement l'influence sur l'ensemble des paramètres. C'est la distance, au sens de la métrique définie par l'inverse de la covariance des paramètres, entre le vecteur des paramètres \mathbf{b} estimé avec toutes les observations et celui estimé lorsque la i ème observation est supprimée.

$$D_i = \frac{1}{2}(\mathbf{b} - \mathbf{b}_{(i)})'(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}(\mathbf{b} - \mathbf{b}_{(i)}).$$

Cet indicateur prend simultanément en compte l'effet levier et l'importance du résidu de chaque observation. Le graphe de ces valeurs est donc plus synthétique et interprétable en tenant compte du graphe des résidus et de celui des termes diagonaux de \mathbf{H} .

6 Compléments

6.1 Sur-dispersion

Dans certaines situations, par exemple lors d'observations dépendantes, la variance de la variable Y_i supposée binomiale ou de Poisson, qui est théoriquement fixée par le

modèle, est plus importante, multipliée par un facteur d'échelle (scale parameter) σ^2 . Si ce paramètre est plus grand que 1, on dit qu'il y a sur-dispersion. Une méthode basée sur une maximisation de la formule de *quasi-vraisemblance* est alors utilisée pour estimer à la fois σ et β .

6.2 Variable “offset”

Lorsque la variable à expliquer dans le cas d'un modèle linéaire généralisé dépend également *linéairement* d'une autre variable, cette dernière est déclarée *offset* et sert ainsi à “tarer” le modèle. Exemple : pour modéliser le nombre de sinistres déclarés par catégorie de conducteurs, la variable *nombre de contrats* est déclarée “offset”.

Annexe B

Introduction au bootstrap

1 Introduction

La motivation du *bootstrap*¹ (Efron, 1982 ; Efron et Tibshirani, 1993) est d’approcher par simulation (*Monte Carlo*) la distribution d’un estimateur lorsque l’on ne connaît pas la loi de l’échantillon ou, plus souvent lorsque l’on ne peut pas supposer qu’elle est gaussienne. L’objectif est de remplacer des hypothèses probabilistes pas toujours vérifiées ou même invérifiables par des simulations et donc beaucoup de calcul.

Le principe fondamental de cette technique de rééchantillonnage est de substituer à la distribution de probabilité inconnue F , dont est issu l’échantillon d’apprentissage, la distribution empirique \hat{F} qui donne un poids $1/n$ à chaque réalisation. Ainsi on obtient un échantillon de taille n dit *échantillon bootstrap* selon la distribution empirique \hat{F} par n tirages aléatoires avec remise parmi les n observations initiales.

Il est facile de construire un grand nombre d’échantillons bootstrap sur lesquels calculer l’estimateur concerné. La loi simulée de cet estimateur est une approximation asymptotiquement convergente sous des hypothèses raisonnables² de la loi de l’estimateur. Cette approximation fournit ainsi des estimations du biais, de la variance, donc d’un risque quadratique, et même des intervalles de confiance de l’estimateur sans hypothèse (normalité) sur la vraie loi.

1.1 Principe du *plug-in*

Soit $\mathbf{x} = \{x_1, \dots, x_n\}$ un échantillon de taille n issue d’une loi inconnue F sur (Ω, \mathcal{A}) . On appelle *loi empirique* \hat{F} la loi discrète des singletons (x_1, \dots, x_n) affectés des poids $1/n$:

$$\hat{F} = \sum_{i=1}^n \delta_{x_i}.$$

¹Cette appellation est inspirée du baron de Münchhausen (Rudolph Erich Raspe) qui se sortit de sables mouvants par traction sur ses *tirants de bottes*. En France “bootstrap” est parfois traduit par *à la Cyrano* (acte III, scène 13) en référence à ce héros qui prévoyait d’atteindre la lune en se plaçant sur une plaque de fer et en itérant le jet d’un aimant.

²Échantillon indépendant de même loi et estimateur indépendant de l’ordre des observations.

Soit $A \in \mathcal{A}$, $P_F(A)$ est estimée par :

$$\widehat{(P)}_F(A) = P_{\widehat{F}}(A) = \sum_{i=1}^n \delta_{x_i}(A) = \frac{1}{n} \text{Card } x_i \in A.$$

De manière plus générale, soit θ un paramètre dont on suppose que c'est une fonction de la loi F . on écrit donc $\theta = t(F)$. Par exemple, $\mu = E(F)$ est un paramètre de F suivant ce modèle. Une *statistique* est une fonction (mesurable) de l'échantillon. Avec le même exemple :

$$\widehat{\mu} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

et \bar{x} est la statistique qui estime μ . On dit que c'est un estimateur "plug-in" et, plus généralement,

DÉFINITION B.1. — On appelle *estimateur plug-in* d'un paramètre θ de F , l'estimateur obtenu en remplaçant la loi F par la loi empirique :

$$\widehat{\theta} = t(\widehat{F}).$$

comme dans le cas de l'estimation de μ : $\widehat{\mu} = E(\widehat{F}) = \bar{x}$.

1.2 Estimation de l'écart-type de la moyenne

Soit X une variable aléatoire réelle de loi F . On pose :

$$\mu_F = E_F(X), \quad \text{et} \quad \sigma_F^2 = \text{Var}_F(X) = E_F[(X - \mu_F)^2];$$

Ce qui s'écrit :

$$X \sim (\mu_F, \sigma_F^2).$$

Soit (X_1, \dots, X_n) n variables aléatoires i.i.d. suivant aussi la loi F . Posons $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$. Cette variable aléatoire a pour espérance μ_F et pour variance σ_F^2/n . On dit aussi que la statistique

$$\bar{X} \sim (\mu_F, \sigma_F^2/n).$$

Remarquons qu'en moyennant plusieurs valeurs ou observations, on réduit la variance inhérente à une observation. De plus, sous certaines conditions sur la loi F et comme résultat du théorème de la limite centrale, \bar{X} converge en loi vers la loi normale.

L'estimateur plug-in de σ_F est défini par :

$$\begin{aligned} \widehat{\sigma}^2 &= \widehat{\sigma_F}^2 = \sigma_{\widehat{F}}^2 = \text{Var}_{\widehat{F}}(X) \\ &= E_{\widehat{F}}[(X - E_{\widehat{F}}(X))^2] = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2. \end{aligned}$$

L'estimateur plug-in de σ_F est (légèrement) différent de celui du maximum de vraisemblance. L'estimateur plug-in est en général biaisé mais il a l'avantage d'être simple et de pouvoir s'appliquer à tout paramètre θ même lorsque l'on ne peut pas calculer la vraisemblance du modèle.

2 Estimation bootstrap d'un écart-type

Soit $\hat{\theta} = s(x)$ un estimateur quelconque (M.V. ou autre) de θ pour un échantillon x donné. On cherche à apprécier la précision de $\hat{\theta}$ et donc à estimer son écart-type.

2.1 Échantillon bootstrap

Avec les mêmes notation, \hat{F} est la distribution empirique d'un échantillon $\mathbf{x} = \{x_1, \dots, x_n\}$.

DÉFINITION B.2. — On appelle échantillon bootstrap de \mathbf{x} un échantillon de taille n noté

$$\mathbf{x}^* = \{x_1^*, \dots, x_n^*\}$$

suivant la loi \hat{F} ; \mathbf{x}^* est un ré-échantillon de \mathbf{x} avec remise.

2.2 Estimation d'un écart-type

DÉFINITION B.3. — On appelle estimation bootstrap de l'écart-type $\widehat{\sigma}_F(\hat{\theta})$ de $\hat{\theta}$, son estimation plug-in : $\sigma_{\hat{F}}(\hat{\theta})$.

Mais, à part dans le cas très élémentaire où, comme dans l'exemple ci-dessus, θ est une moyenne, il n'y a pas de formule explicite de cet estimateur. Une approximation de l'estimateur bootstrap (ou plug-in) de l'écart-type de $\hat{\theta}$ est obtenue par une simulation (Monte-Carlo) décrite dans l'algorithme ci-dessous.

Pour un paramètre θ et un échantillon \mathbf{x} donnés, on note $\hat{\theta} = s(\mathbf{x})$ l'estimation obtenue sur cet échantillon. Une *réplication bootstrap* de $\hat{\theta}$ est donnée par : $\hat{\theta}^* = s(\mathbf{x}^*)$.

ALGORITHME B.1 : Estimation bootstrap de l'écart-type

- Soit \mathbf{x} un échantillon et θ un paramètre.
- Pour $b = 1$ à B Faire
 - Sélectionner 1 échantillon bootstrap $\mathbf{x}^{*b} = \{x_1^{*b}, \dots, x_n^{*b}\}$. par tirage avec remise dans x .
 - Estimer sur cet échantillon : $\hat{\theta}^*(b) = s(\mathbf{x}^{*b})$.
- Fin pour
- Calculer l'écart-type de l'échantillon ainsi construit :

$$\begin{aligned} \hat{\sigma}_B^2 &= \frac{1}{B-1} \sum_{b=1}^B (\hat{\theta}^*(b) - \hat{\theta}^*(.))^2 \\ \text{avec } \hat{\theta}^*(.) &= \frac{1}{B} \sum_{b=1}^B \hat{\theta}^*(b). \end{aligned}$$

$\hat{\sigma}_B$ est l'approximation bootstrap de l'estimation plug-in recherchée de l'écart-type de $\hat{\theta}$.

2.3 Estimation du biais

Avec les mêmes notations :

$$\theta = t(F) \quad \text{et} \quad \hat{\theta} = s(\mathbf{x}),$$

le biais d'un estimateur s'exprime comme

$$\mathcal{B}_F(\hat{\theta}) = E_F[s(\mathbf{x})] - t(F).$$

Un estimateur est sans biais si $E[\hat{\theta}] = \theta$. Le biais est aussi une mesure de la précision d'un estimateur et on a vu que, généralement, les estimateurs plug-in étaient biaisés.

DÉFINITION B.4. — On appelle *estimateur bootstrap du biais*, l'*estimateur plug-in* :

$$\widehat{\mathcal{B}}_F(\hat{\theta}) = \mathcal{B}_{\hat{F}}(\hat{\theta}) = E_{\hat{F}}[s(\mathbf{x}^*)] - t(\hat{F}).$$

Comme pour l'écart-type, il n'existe généralement pas d'expression analytique et il faut avoir recours à une approximation par simulation.

ALGORITHME B.2 : Estimation bootstrap du biais

- Soit \mathbf{x} un échantillon et θ un paramètre.
 - Pour $b = 1$ à B Faire
 - Sélectionner 1 échantillon bootstrap $\mathbf{x}^{*b} = \{x_1^{*b}, \dots, x_n^{*b}\}$. par tirage avec remise dans \mathbf{x} .
 - Estimer sur cet échantillon la réplication bootstrap de $\hat{\theta}$: $\hat{\theta}^*(b) = s(\mathbf{x}^{*b})$.
 - Fin pour
 - Approcher $E_{\hat{F}}[s(\mathbf{x}^*)]$ par $\hat{\theta}^*(.) = \frac{1}{B} \sum_{b=1}^B \hat{\theta}^*(b)$
 - L'approximation bootstrap du biais est : $\widehat{\mathcal{B}}_B(\hat{\theta}) = \hat{\theta}^*(.) - \hat{\theta}$.
-

3 Compléments

En résumé, on peut dire que le bootstrap repose sur une hypothèse très élémentaire : $\hat{\theta}^*$ se comporte par rapport à $\hat{\theta}$ comme $\hat{\theta}$ par rapport à θ . La connaissance de $\hat{\theta}^*$ (distribution, variance, biais...) renseigne alors sur celle de $\hat{\theta}$.

Beaucoup d'autres compléments sont à rechercher dans la littérature et en particulier dans Efron et Tibshirani (1993). Il est ainsi possible de définir des intervalles de confiance bootstrap en considérant la distribution et les quantiles de $\hat{\theta}^*$ ou même encore des tests à partir des versions bootstrap de leur statistique.

Le bootstrap rapidement décrit ici est dit "non-paramétrique" car la loi empirique \hat{F} est une estimation non-paramétrique de F . Dans le cas où F serait connue à un paramètre près, il existe également une version dite *paramétrique* du bootstrap.

Pour des estimateurs plus compliqués (fonctionnels) comme dans le cas de la régression non-paramétrique par noyau ou spline, il est facile de construire graphiquement une enveloppe bootstrap de l'estimateur à partir de réplifications de l'échantillon. Celle-ci fournit généralement une bonne appréciation de la qualité de l'estimateur obtenu. Attention, dans

le cas de la régression il est en principe plus justifié de répliquer le tirage sur les *résidus* plutôt que sur les observations. Ce sont les résidus qui sont en effet supposés i.i.d. et qui vérifient donc les hypothèses nécessaires mais cette approche devient très sensible à l'hypothèse sur la validité du modèle. Il est finalement d'usage de considérer un échantillon bootstrap issu des données initiales (Efron et Tibshirani) :

$$\mathbf{z}^{*b} = \{(\mathbf{x}_1^{*b}, y_1^{*b}), \dots, (\mathbf{x}_n^{*b}, y_n^{*b})\};$$

c'est ce qui a été choisi dans ce document.

Enfin, l'estimation bootstrap est justifiée par des propriétés asymptotiques (convergence en loi) lorsque le nombre de réplifications (B) croît conjointement avec la taille de l'échantillon (n).

Table des matières

	Motivations du <i>data mining</i>	3
	Stratégie du <i>data mining</i>	4
	Objectif	5
1	Introduction	7
1	Objectif	7
2	Problématique	7
2.1	Supervisé <i>vs.</i> non-supervisé	7
2.2	Modélisation <i>vs.</i> apprentissage	8
2.3	Discrimination <i>vs.</i> régression	8
2.4	Statistique, informatique et taille des données	8
2.5	Choix de méthode	9
2.6	Choix de modèle : équilibre biais-variance	9
2.7	Choix de modèle : sélection <i>vs.</i> régularisation	10
2.8	Contenu	10
2	Régression linéaire	13
1	Introduction	13
2	Modèle	13
3	Estimation	14
3.1	Estimation par M.C.	14
3.2	Propriétés	15
3.3	Sommes des carrés	16
3.4	Coefficient de détermination	16
4	Inférences dans le cas gaussien	16
4.1	Inférence sur les coefficients	16
4.2	Inférence sur le modèle	17
4.3	Inférence sur un modèle réduit	17
4.4	Prévision	18
4.5	Exemple	18

5	Choix de modèle	20
5.1	Critères	20
5.2	Algorithmes de sélection	22
5.3	Exemple	23
5.4	Choix de modèle par régularisation	24
6	Compléments	27
6.1	Modèles curvilinéaires	27
6.2	Influence, résidus, validation	27
7	Analyse de variance à un facteur	30
7.1	Introduction	30
7.2	Modèle	31
7.3	Test	33
8	Analyse de covariance	34
8.1	Modèle	34
8.2	Tests	35
8.3	Choix de modèle	36
8.4	Exemple	36
9	Introduction	39
10	Odds et odds ratio	39
11	Régression logistique	40
11.1	Type de données	40
11.2	Modèle binomial	41
12	Choix de modèle	42
12.1	Recherche pas à pas	42
13	Exemples	42
13.1	Débits×Volumes	42
13.2	Données bancaires	44
3	Erreur de prédiction	47
1	Introduction	47
2	Erreur de prédiction	47
2.1	Définition	47
2.2	Décomposition	48
2.3	Estimation	48
3	Estimation avec pénalisation	49
3.1	C_p , AIC, BIC	49
3.2	Dimension de Vapnik-Chernovenkis	50

4	Estimation par simulation	51
4.1	Validation croisée	51
4.2	Bootstrap	51
4.3	Remarques	53
4	Analyse Discriminante Décisionnelle	55
1	Introduction	55
2	Règle de décision issue de l'AFD	56
2.1	Cas général : m quelconque	56
2.2	Cas particulier : $m = 2$	56
3	Règle de décision bayésienne	56
3.1	Introduction	56
3.2	Définition	57
3.3	Coûts inconnus	57
3.4	Détermination des <i>a priori</i>	57
3.5	Cas particuliers	58
4	Règle bayésienne avec modèle normal	58
4.1	Hétéroscédasticité	58
4.2	Homoscédasticité	59
4.3	Commentaire	59
5	Règle bayésienne avec estimation non paramétrique	59
5.1	Introduction	59
5.2	Méthode du noyau	60
5.3	k plus proches voisins	60
6	Exemple	61
5	Arbres binaires	63
1	Introduction	63
2	Construction d'un arbre binaire	63
2.1	Principe	63
2.2	Critère de division	65
2.3	Règle d'arrêt	65
2.4	Affectation	65
3	Critères d'homogénéité	65
3.1	Y quantitative	66
3.2	Y qualitative	67
4	Élagage	68
4.1	Construction de la séquence d'arbres	71

4.2	Recherche de l'arbre optimal	71
6	Méthodes connexionistes	73
1	Historique	73
2	Réseaux de neurones	74
2.1	Neurone formel	74
3	Perceptron multicouche	75
3.1	Architecture	75
3.2	Apprentissage	76
3.3	Utilisation	77
7	Agrégation de modèles	79
1	Introduction	79
2	Famille de modèles aléatoires	79
2.1	<i>Bagging</i>	79
2.2	Forêts aléatoires	81
3	Famille de modèles adaptatifs	82
3.1	Principes du <i>Boosting</i>	82
3.2	Algorithme de base	83
3.3	Version aléatoire	84
3.4	Pour la régression	84
3.5	Modèle additif pas à pas	85
3.6	Régression et boosting	86
3.7	Compléments	87
4	Application	89
4.1	Logiciels	89
4.2	Résultats comparatifs	89
A	Introduction au modèle linéaire général	95
1	Composantes des modèles	95
1.1	Distribution	95
1.2	Prédicteur linéaire	96
1.3	Lien	96
1.4	Exemples	97
2	Estimation	98
2.1	Expression des moments	98
2.2	Équations de vraisemblance	98
2.3	Fonction lien canonique	99

3	Qualité d'ajustement	100
3.1	Déviance	100
3.2	Test de Pearson	100
4	Tests	101
4.1	Rapport de vraisemblance	101
4.2	Test de Wald	101
5	Diagnostics	102
5.1	Effet levier	102
5.2	Résidus	102
5.3	Mesure d'influence	103
6	Compléments	103
6.1	Sur-dispersion	103
6.2	Variable "offset"	104
B	Introduction au bootstrap	105
1	Introduction	105
1.1	Principe du <i>plug-in</i>	105
1.2	Estimation de l'écart-type de la moyenne	106
2	Estimation bootstrap d'un écart-type	107
2.1	Échantillon bootstrap	107
2.2	Estimation d'un écart-type	107
2.3	Estimation du biais	108
3	Compléments	108