N° d'ordre :17/2016-M/MT

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère De L'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediene

Faculté des Mathématiques

MEMOIRE

Présenté pour l'obtention du diplôme de MAGISTER

En : MATHEMATIQUE

Spécialité : Probabilités - Statistiques

Par : **BOUCHAFAA Asma**

Sujet

Étude comparative de modèles de prévision sur données *d'enquêtes*

.

Soutenu publiquement, le 9/05/2015, devant le jury composé de :

M^me^ Djaballah khedidja	Professeur à l'USTHB	Présidente
M Djedour Mohamed	Professeur à l'USTHB	Directeur de mémoire
M^me^ Saggou Hafida	Maître de Conférence/A à l'USTHB	Examinatrice
M Medkour Tarek	Maître de Conférence/A à l'USTHB	Examinateur

Remerciements

Je tiens à remercier en premier lieu mon directeur de mémoire, M DJEDOUR Mohamed, Professeur à l'USTHB pour avoir accepté de m'encadrer en mémoire de Magister, pour ses nombreux conseils, mais surtout pour sa disponibilité et sa patience.

Je remercie sincèrement Mme DJABALLAH Khedidja, Professeur à l'USTHB, qui m'a fait l'honneur en acceptent de présider le jury de soutenance de mon mémoire.

Je remercie également Melle Saggou Hafida, MCA à l'USTHB et Mr MEDKOUR Toufik,MCA l'USTHB pour l'honneur qu'ils me font en acceptent de faire partie du jury.

Je tiens à remercier plus personnellement mon marie, mes filles , mes parents, mes frères et mes sœurs pour le soutien et l'encouragement qu'ils m'ont apportés durant toutes mes années d'études, ainsi que tous ceux qui m'ont de prés ou de loin aidée.

Je dédie ce modeste travail à :

Mon mari , Belaroussi Samir,

Mon père , BOUCHAFAA BelKACEM,(allah yarahim)tu es toujours dans mon coeur
Ma mère ,KHERRAT Fatima,

Mes filles,Sirine et Sabrina, Hanane

Mon beau père , Belaroussi mouloud,
Ma belle mère ,Bouchafa Nouara
Mes soeurs : louiza et Karima Safia Rafika,

Mes frères :kamel Sofiane Badr Abderahim Djeber Oussama,

A toute ma famille Bouchafaa et Belaroussi,

A tous mes chères amis (Amira , Meriem pour leur amitié, leurs encouragement, leur soutien.

Table des matières

Table des figures

Liste des tableaux

LISTE DES ABRÉVIATIONS

BPCO : Broncho-pneumopathies obstructives chroniques

CHU : Centre hospitalo-universitaire

CIM10 : Classification Internationale des Maladies “10”

CREAD : Centre de Recherche en Economie Appliquée et Développement

CSP : Catégories socio-professionelles

CV : Maladies cardio-vasculaire

EHS : Etablissement hospitalier spécialisé

GCT : Glucose - Cholestérol - Triglycérides

HTA : Hypertension artérielle

IDF : Fédération Internationale du Diabète

IMC : Indice de masse corporelle

INCO MED : International Scientific Coopération Projects

INSP : Institut National de Santé Publique

IRD : Institut de Recherche et de Développement (U106 Nutrition Alimentation et Société Montpellier)

antediab : Antécédents familiaux de diabète PA : Pression artérielle

PAD : Pression artérielle diastolique

PAS : Pression artérielle systolique

TA : Tension artérielle

TAHINA : Transition and Health Impact In North Africa

UGD : Ulcère gastro-duodénal

US NCEP ATP III : United States National Cholesterol Education Program Adult Treatment Panel III

INTRODUCTION

La transition sanitaire est un concept développé par Omran au début des années 60 pour expliquer l'évolution de la situation épidémiologique et son interaction avec les facteurs démographiques, environnementaux et socio-économiques constatées au cours des 18ème et 19ème siècles dans les pays occidentaux et au 20ème siècle dans les pays du sud. Ce concept a évolué au cours du temps au vu de situations observées non prises en compte au départ.
L' Algérie, pays émergeant, à l'instar de tous les pays du monde, traverse depuis maintenant une vingtaine d'années une transition sanitaire révélée par différentes études. En effet « l'âge de la pestilence et de la famine » est largement dépassé ; les maladies transmissibles et les problèmes de santé maternelle et infantile ont sensiblement diminué grâce à l'amélioration des conditions de vie et de la couverture sanitaire et la mise en oeuvre de programmes nationaux de santé publique.
Ce qui a eu pour conséquence une baisse notable de la mortalité générale.
Ceci s'est accompagné d'une augmentation progressive de l'espérance de vie (de moins de 50 ans en 1962 à 74,6 ans en 2005) et d'une transition démographique plus tardive (années 1980) qui s'est manifestée par une modification de l'aspect de la pyramide des âges dans laquelle la proportion des populations les plus jeunes (moins de 20 ans) amorce une diminution progressive alors que celle des populations adultes est en nette augmentation (les 20 - 59 ans représentent $41,5\%$ de la population et$7,1\%$ ont 60 ans et plus en2005 et arriveront à 10% en 2015) ; ce qui a corollaire un vieillissement progressif de la population et l'augmentation du poids des maladies chroniques que des études effectuées lors de la décennie écoulée ont identifié, notamment l'enquête nationale santé réalisée par l'INSP
Cette enquête a mis en évidence que les affections les plus fréquentes sont les maladies cardiovasculaires, les maladies respiratoires, les maladies ostéoarticulaires et le diabète [ENS, INSP, 1990].
Plus récemment, l'étude relative à l'analyse des causes de décès en population générale dans un échantillon de 16 wilayas et l'analyse des motifs d'hospitalisation dans un échantillon d'établissements hospitaliers répartis sur le territoire national [TAHINA, INSP, 2002] révèlent que les affections chroniques suivantes occupent dans notre pays une place prépondérante dans la charge de morbidité actuelle :

- L'hypertension artérielles et ses complications vasculaires (cardiaques, cérébrales, artériels périphériques)

- Le diabète sucré
- Les affections respiratoires chroniques (asthme, bronchite chronique, broncho-pneumopathie chronique obstructive)
- Les maladies digestives (ulcères digestifs, lithiase biliaire, colopathies)
- L'insuffisance rénale chronique
- Les cancers
- Les maladies mentales.

Les déterminants de ces affections sont aujourd'hui'hui en majorité connus, notamment le diabétique et les facteurs liés aux modes de vie et aux comportements des individus tels que la sédentarité, L'évolution des habitudes alimentaires, l'usage du tabac et la consommation d'alcool dont il convient d'en déterminer l'importance et l'impact sur la santé des algériens.
Et à cet effet l'INSP a réalisé en juin 2005 une enquête nationale santé qui rentre dans le cadre global d'un projet de recherche sur la transition épidémiologique et son impact sur la santé dans les pays nord africains [TAHINA] et dont les objectifs sont :

- L'estimation de la morbidité au niveau de la population
- L'estimation de la consommation de soins
- L'estimation de la fréquence des facteurs de risques chez les adultes de 35 à 70 ans.

Le Projet TAHINA « Transition Epidémiologique et Impact sur la Santé en Afrique du Nord » est un projet de recherche financé par l'union Européenne dans le cadre du programmeINCO « Confirming the international Role of Community Research » volet INCO-MED.[**1**]
Il part du principe que la transition épidémiologique, caractérisée par la persistance ou la ré émergence des « maladies du passé » et l'augmentation de l'importance des maladies chroniques, pose de façon accrue la problématique des (la) stratégie(s) d'intervention sanitaire à lancer sur le terrain. Les actions engagées par l'Institut National de Santé Publique dans le cadre du projet TAHINA sont des tentatives de réponse visant l'élaboration de recommandations à l'attention des acteurs du système de santé impliqués dans la gestion de cette transition.
Lesquelles recommandations, réunies avec d'autres données sanitaires issues d'autres sources, apporteront une contribution à la réorientation du système de santé amorcée déjà depuis quelques mois dans le cadre des réformes sanitaires.
Afin d'asseoir les (la) stratégie(s) d'intervention, le projet se propose d'articuler deux types d'analyse complémentaires :

- La caractérisation de la transition épidémiologique,de ses déterminants et de ces conséquences.

- L'analyse des représentations de cette transition par les acteurs (populations, professionnels et décideurs) et l'analyse des pratiques qui s'y rapportent.

Les objectifs généraux du projet visent à :

- renforcer la capacité des services de santé à gérer les problèmes posés par l'avancée de la transition épidémiologique à travers une stratégie globale, intégrée et multisectorielle ;
- augmenter l'attention à la prévention des maladies chroniques non transmissibles de tous les secteurs concernés par les changements dans les modes de vie .

  Les objectifs spécifiques se proposent de :

- Mesurer la charge de morbidité globale et ses coûts associés liés à la transition Épidémiologique ;

- Caractériser les déterminants alimentaires, économiques, sociaux, culturels et Environnementaux de cette situation.

- les représentations et l'évolution des pratiques actuelles des professionnels de santé face aux changements de la situation sanitaire et nutritionnelle ;

- Identifier la perception et la sensibilité des acteurs d'autres secteurs concernés sur ces changements

- Mettre en évidence l'évolution des représentations et des pratiques de la population en matière de santé, d'alimentation et de modes de vie ;

- Initier un processus d'élaboration conjoint de stratégies d'interventions intégrées et globales.

  Quatre grands hèmes, organisés en work packages, ont été choisis à cet effet :
  WP1 : Evaluation de la charge de morbidité qui comprend les thèmes d'étude suivants :

- Évaluation des causes de mortalité

- Appréciation de la morbidité au niveau hospitalier

- Estimation de la morbidité, de la consommation de soins et des facteurs de risques de maladies chroniques au niveau de la population

- Analyse des coûts

  WP2 : Caractérisation de l'environnement socio-économique et des modes de vie qui comprend les thèmes d'étude suivants :

- Cadre conceptuel des changements liés à la transition

- Analyse de l'environnement socio-économique

– Évaluation des profils de consommation alimentaire et d'activité physique

WP3 : Analyse des attitudes et pratiques des acteurs qui comprend les thèmes d'étude suivants

– Etude de groupes de population

– Enquête auprès des personnels de santé

– Etude des processus de décision en liaison avec les décideurs
WP4 : Elaboration de stratégies qui comprend les thèmes d'étude suivants

– Rétrospective des interventions connues pour juger de leur degré d'adaptation possible

– Réflexion méthodologique sur l'élaboration des stratégies

– Intégration progressive des différents résultats de recherche

– Contribution à la réflexion sur les stratégies au niveau national.

Le but de cette recherche est d'évaluer des modèles de prévision basés sur les arbres de décision ,la régression logistique et les réseaux de neurones pour la réalisation d'un système d'aide à la décision sur des données et de comparer leurs performances. Les utilisateurs d'un tel système d'aide à la décision seront des décideurs, pour la plupart experts en analyse financière, ou responsables du risque de crédit dans les banques, ou superviseurs bancaires ou des médecins cherchant à établir un diagnostic. Dans notre cas on s'intéressera à certaines maladies décrites dans l'enquête TAHINA réalisée en 2005 par l'Institut National de la Santé Publique et aux relations causales possibles entre elles telles que suggérées par des spécialistes. On s'intéressera à certaines maladies à prévoir au vu des symptômes relevés en particulier le diabète. Les données seront toutes de nature catégorielles(ou transformées en données catégorielles). Afin d'effectuer le travail demande INSP a mis à notre disposition une base de données constituée de 1312 variables sur un échantillon 4818 individus Mais notre base de données obtenue de TAHINA constitue l'outil de recueil d'informations qui permet de collecter grâce à un grand nombre de questions des renseignements relatifs à notre mode de vie et notre culture et nos habitudes alimentaires La vérification des 1312 questionnaires (variables) renseignés et réceptionnés, a abouti à garder 254 variables sur un échantillon de 4818 individus en raison du nombre élevé de questions non renseignées, d'incohérence de certaines réponses les rendant ainsi inexploitables ainsi que le manque de certaines valeurs. A l'aide de logiciel R, Pour chaque variable choisie de TAHINA dépassant 10% de données manquantes, cette variable est automatiquement exclue de l'étude. La présente étude s'articule sur une base de données au total de 254 variables dont :

54 variables quantitatives ;

200 variables qualitatives.

Trois applications informatiques sous R ont été appliquées à savoir :

La régression logistique
Pour les variables qualités deux modalités le package (glm)
Pour les variables qualités plus que deux modalités (polytomique) le package (vgam)
Les arbres de décision par le package (rpart)
Les réseaux de neurones le package (nnet)

Les variables sont scindées en (12) douze groupes pour les raisons ci-après : Les différentes volets du questionnaire sont alors passées à la tirée au sort à savoir :[1]

Volet A

« aspects socioéconomique - morbidité - facteurs de risque »
Volet B

« nutrition : alimentation consommées- fréquence de consommation alimentaire ?pratique alimentaire »
Volet C

« qualité de vie »

Pour le Module I : « identification , conditions de vie et caractérisation du ménage » du volet A ainsi que le module II : « recours aux soins et morbidité dans le ménage » la personne enquêtée est le chef du ménage.

Les autres modules (module III,IV,V) du volet A, la volet B, le volet C porte sur le membre du ménage tiré au sort dont l ?âge est supérieur ou égal à 35 ans et inférieur ou égal à 70 ans .

Pour le Module$I$: « identification , conditions de vie et caractérisation du ménage » du volet A ainsi que le module II : « recours aux soins et morbidité dans le ménage » la personne enquêtée est le chef du ménage.

Les autres modules (module III,IV,V) du volet$A$, la volet B, le volet C porte sur le membre du ménage tiré au sort dont l'âge est supérieur ou égal à 35 ans et inférieur ou égal à 70 ans.

Groupe 1 : ce groupe contient 21 variables quantitatives concernant le cumul de consommation par semaine de certains alimentation.
« painjc, couscouljc, patejc, rizjc, legumejc, fruitjc, pomterrjc, "legusecjc, laitjc, poissonjc, viandejc, volailljc,oeufjc, cacahuejc, dessertjc, huilolijc, huilejc, beurrejc, feculenj, âge , cerealej».
Groupe 2 : ce groupe contient28 variables qualitatives « consommation de fruits » :
Groupe 3 : groupe des légumes contient 35 variables qualitatives
Groupe 4 : ce groupe contient 14 variables concernant les produits laitiers.
Groupe 5 : ce groupe contient 18 variables qualitatives des différents types des protéines (viandes et volailles et ...)
Groupe 6 : ce groupe contient 23 variables qualitatives des légumes et fruits secs
Groupe7 : le groupe contient des variables qualitatives 17 concernant les différents types de gâteaux et les sucres.
Groupe 8 : contient des variables qualitatives(dicthomies et polytomiques) concernant la description d'un individu
Groupe 9 : ce groupe contient des variables concernant le lieu et la convivialité des repas.
Groupe 10 : contient des variables qualitatives concernant les gâteaux et les types de boissons
Groupe 11 : ce groupe contient des variables concernant les habitudes toxiques.
Groupe 12 :ce groupe contient des variables concernant le loisir (passe-temps)

Notre travail s'articule sur une idée directrice, en l'occurrence l'étude comparative de modèles de prévision sur données d'enquêtes à l'aide des modèles de régression logistique, les arbres de décisions et de réseaux de neurones de type multicouches, dont le principe consiste à utiliser les données et un critère à minimiser pour réaliser un modèle.

Dans le premier chapitre nous allons résumer ex ante la démarche du statisticien lors du dépouillement d'une enquête sur ordinateur avec les outils logiciel disponibles actuellement et les étapes du traitement, on traitera les données manquantes ex post.

Nous avons pris en considération les différents types de valeurs manquantes, la méthode choisie. D'autres méthodes ont été utilisées à l'instar de la plupart des propriétés mathématiques et statistiques des composantes principales basées sur la matrice de covariance (ou de corrélation) connue d'une population.

Dans le deuxième chapitre, nous allons développer les outils de modélisation de régression logistique, tout en approfondissant les notions des modèles logistiques utilisés en particulier dans la prévision.

Dans le troisième chapitre, nous allons exposer un rappel des outils de base de l'analyse de régression logistique, et nous mettrons l'accent sur les modèles arbres de décision.

Dans le cinquième chapitre nous allons présenter le modèle neuronal le plus connu qui est la Perceptron multicouches (PMC). Nous rappellerons dans le premier paragraphe les propriétés théoriques associées au PMC, qui expliquent mieux pourquoi l'utilisation d'un PMC permet souvent d'obtenir des performances intéressantes.

Nous allons clôturer notre mémoire par une conclusion générale et des perspectives de recherche.

CHAPITRE

RAPPELLES ET GÉNÉRALITÉS

I. Traitement des données d'enquête

1. introduction

Projet TAHINA[1]

Une enquête désigne dans la langue courante une investigation qui se propose de recueillir ou de rassembler des informations sur un thème donnée, le plus souvent a partir d'interrogations, d'interrogatoires, entretenus, d'entretien, ou, dans certains cas, simplement a partir de documents ou d'indices. Si le sens usuel du mot enquête. Lorsque l'on parle d'enquête policières, d'enquête réalisé par les journalistes, ces enquêtes ne supposent pas l'existence d'une population, ni l'établissement d'un recueil de données codifier, normalisées.

Pour les statisticiens, le mot enquête désigne le plus souvent une enquête par sondage, les enquêtes démographiques, socioéconomique, sociopolitique, épidémiologique, les enquêtes audiométriques ou de marketing appartiennent a cette catégorie. La forme canonique du fichier d'enquête sera pour nous le fichier rectangulaire de dimension (n, p), dont les$p$ colonnes représentent les variables pouvant être numérique (mesure, notes), nominales (numéros repérant) des catégories, alors que les n lignes représentent des individus, des observations, des objets, des parcelles de terrain, des événements.

Ces éléments lignes qui peuvent être divers sont décrits par des variables relatives a différents thème, ce qui permet de multiplier les points de vue, de les confronter par des tris, des graphiques, de les relier par des calculs, de coefficient ou de modèles particuliers. Les outils récents de calculs permettent d'établir des typologies d'individus, de décrire des associations entre variable, de mettre en oeuvre un savoir-faire. Un savoir -faire particulier que l'on désigne précisément par « Traitement des données d'enquête ».

2. Les étapes du traitement :

Les étapes du traitement :

Résumons la démarche du statisticien lors du dépouillement d'une enquête sur ordinateur avec les outils logiciels disponibles actuellement.
Le dépouillement d'enquête traditionnel met en oeuvre des techniques simples, éprouvées , facile à interpréter :les tris ,les tableaux croisées, c'est-à-dire des calculs de pourcentages d'individus pour chaque modalité d'une variable nominale(ces pourcentages seront calculés par rapport à l'échantillon global, mais aussi par rapport à des sous- échantillons) et des calculs de moyennes de variables numériques ou quantitatives (qui peuvent être ventilées selon les catégories d'une ou de plusieurs variables nominales).
Des méthodes statistiques plus élaborées viennent parfois compléter ces premiers résultats : Régression, analyse de la variance, modèles log-linéaires.
Des méthodes statistiques d'analyse des données (analyse descriptives multidimensionnelles) modifient profondément les premières phases du traitement des données d'enquête.
Elles vont en fait bouleverser l'enchaînement des taches, et définir une nouvelle méthodologie. Dans le cadre de cette méthodologie, les étapes du traitement des données d'enquêtes sont brièvement, les suivantes :
1-descriptions élémentaires (histogrammes, moyennes, ecart-types, valeurs extrême, quantiles).retour éventuel aux données de base pour des corrections.
2-conjecture sur les non-réponses :
L'erreur de la non-réponse c'est-à-dire l'erreur due à l'absence total ou partielle d'informations concernant des individus de l'échantillon.le problème du aux non-réponses partielles et généralement moins aigu :
Les réponses aux autres questions du questionnaire donnent souvent des pistes d'explication exploitées par les méthodes d'imputation.
3-épreuves d'hypothèse classique (modèle log linéaire, test statistique usuel, régression, analyse de la variance).
Ces opérations, intervenant au début de la chaîne de traitement, permettent de piloter la suite du dépouillement de l'enquête .le choix des modèles n'est fait de façon aveugle en fonction des hypothèses de base : ces hypothèses pourront souvent être critiquées, d ?autres hypothèse pourront être suggérées.
Les outils élémentaires tout d'abord les principes communs à toutes les méthodes de statistiques descriptives multidimensionnelle.chacune des deux dimensions d'un tableau rectangulaire de données numériques permet de définir des distances (ou des proximités) entre les éléments définissant l'autre dimension : ainsi l'ensemble des colonnes (variables, attribut, mesures) permet de définir, à l'aide de formules appropriés,des distances entre lignes(individus, observation).de la même façon, l'ensemble des lignes permet de calculer des distances entre colonnes. On obtient ainsi des tableaux de distances, qui sont associées à des représentations géométriques complexes.il s'agit de rendre assimilables et accessibles à l'intuition ces représentations, au prix d'une perte d'information de base qui doit rester la plus petite possible.rappelons qu'il existe deux familles de méthodes qui permettent d'effectuer ces réductions :
Les méthodes factorielles.
Les méthodes de classifications.
Ces deux familles de méthodes pourront être utilisées de façon complémentaire pour décrire de la façon la plus exhaustive possible les grands tableaux numériques constitués par les données d'enquêtes.

Les règles d'interprétations des représentations obtenues à l'issue de ces techniques de réduction n'ont pas la simplicité de celles de la statistique descriptive élémentaire. L'analyse des correspondances et l'analyse en composantes principales. Permettent de trouver des sous espaces de représentations des proximités entre profils ou entre vecteurs de description d'observation. Mais elles permettent aussi de positionner dans ce sous espace des lignes ou des colonnes supplémentaires du tableau de données.
On peut ainsi illustre les plans factoriels par des informations n'ayant pas participé à la construction de ces plans,ce qui va avoir des conséquences très importantes au niveau de l'interprétation des résultats.
Complémentarité de la classification :
Dans le cas du traitement statistique des fichiers d'enquêtes en vraie grandeur, la démarche précédente fondée sur des représentations graphiques a deux graves inconvénients :
1-les visualisations sont limitées à deux ou en général à très peu de dimensions, alors que le nombre d'axes significatifs peut souvent atteindre 8 ou 10, pour fixer les idées.
2- ces visualisations peuvent inclure des centaines de points et donner lieu à des graphiques chargés ou illisibles .il faut donc à ce stade faire appel de nouveau aux capacités de gestion et de calcul de l'ordinateur pour compléter, alléger et clarifier la présentation des résultats.
Lorsqu'il a trop de points sur un graphique, il parait utile de procéder à des regroupements fonctionnent de la même façon, que les points soient situés dans un espace à deux ou à dix dimensions.
Une fois les individus regroupés en classes, il est facile d'obtenir une description automatique de ces classes : on peut en effet, pour les variables numériques comme pour les variables nominales, calculer des statistiques d ?écarts entre valeurs internes à la classe et les valeurs globales ; on peut également convertir ces statistiques en valeurs- tests et opérer un tri sur ces valeurs-test. On obtient finalement, pour chaque classe, les modalités et les variables les plus caractéristiques.

II. Traitement des données manquantes

Lebart[2]

Le statisticien est très régulièrement confronté à la problématique des données manquantes, le traitement simultané des données manquantes est souvent nécessaire dans l'analyse des données. Dans le cas d'une enquête lorsque des individus individu n'ont pas répondu à un nombre important de question ; il est possible d'assimiler les observations correspondantes à de la non réponse partielle. Celle-ci est alors est corrigé par repondération des autres observations pour corriger la non-réponse partielle.

1. Les différents types des valeurs manquantes :

On distingue deux catégories de non-réponse :

– La non-réponse totale : lorsque aucune information n'est recueillie sur une

unité échantillonnée.

– La non-réponse partielle : lorsque le manque d'information est limite à certaines variables.

– Des valeurs manquantes parce que :

– n'ont pas pu être observées

– Elles ont été perdues

– Elle était incohérente.

Imputation par la méthode Hot-deck :

Les méthodes de hot-deck consistent à remplacer pour une observation appelée receveur, une valeur manquante sur une variable donnée par une valeur observée sur la même variable un individu répondant appelé donneur .si la population est trop hétérogène pour que le répondant et le non répondant soient « relativement proche » (c'est-à-dire pour que certains de leurs caractéristique soient identique ou proche) ; elle peut être décomposée en sous population plus homogène. Des classes d'imputation sont alors constituées et le donneur est choisi à l'intérieur de la classe à laquelle appartient le receveur.

Imputation indépendante par hot-deck séquentiel :

Dans un premier temps, la non réponse partiel est corrigée par la méthode du hot-deck séquentielle, couramment employée.la méthode du hot-deck séquentielle est relativement simple à mettre en oeuvre.il s'agit de remplacer la valeur manquante par la modalité du répondant précèdent.

Imputation indépendante par hot-deck aléatoire :

Afin de tenir compte des éventuelles relations entre les différentes à imputer, le même donneur est utilisé pour imputer simultanément toutes les variables non renseignées d'une même observation. On utilise la méthode du hot deck aléatoire qui consiste à remplacer une valeur manquante par la valeur observée pour un individu répondant choisit au hasard ; éventuellement à l'intérieur de la classe à laquelle appartient le receveur.

-Imputation simultanée des valeurs prise par le donneur le plus proche du receveur : Le hot-deck métrique est une méthode d'imputation qui consiste à remplacer une valeur manquante pour un receveur par la valeur observée pour le donneur le plus proche, au sens d ?une distance.
La distance est calculée à partir des variables auxiliaires renseignées pour les répondants et les non répondants.

Les méthodes choisies :

Dans notre étude nous avons choisi comme traitement pour les valeurs manquantes quantitatives l'imputation par la moyenne des observations de la variable associé en

suivant les étapes suivantes :
– Tracez l'histogramme de la variable.

– Éliminer un ensemble d'observation afin d'avoir une distribution normal.

– Calculer la moyenne des observations restantes.
Remplacer les valeurs manquantes par la valeur moyenne calculée.

Pour les valeurs manquantes qualitatives nous avons choisi la méthode du hot-deck aléatoire.

2. Les différents traitements des données manquantes :

A. Ne rien faire.

B. Utiliser uniquement l'enregistrement pour les quels les données sont complètes.

C. Utiliser une méthode de repondération

D. Imputer une valeur par des méthodes tells
– moyenne
– ratio
– régression
– Hot-Deck aléatoire
– plus proche voisin

Traitement A : Ne rien faire :
Cela oblige à travailler avec un fichier de données incomplètes qui ressemble à un morceau de fromage gruyère. Si les valeurs manquantes sont peu nombreuses, on peut les oublier sans aucun scrupule

Traitement B : utiliser uniquement les enregistrements complets :

Si les données sont présentées sous forme de tableau, cela revient à oublier une ligne des qu'il manque une valeur dans cette ligne, on oublie donc aussi les autre valeurs de cette ligne qui sont effectivement présentes.
Bien que cette option soit simple et permette d'utiliser un fichier complet, elle prête certain risque. En effet :
– L'échantillon de ceux qui ont répondu à toutes les questions peut être :
– Soit trop réduit pour être significatif
– Soit non représentatif de la population globale
– ne dépend d'aucune des variables d'intérêts.
Cette option ne peut être envisagée que pour une brève analyse descriptive des réponses complètes.

Traitement C : Repondération : Non-réponse- totale : les méthodes de repondération augment
les poids de Sandage applique aux répondants pour compenser pour les non- répondants. L'objectif est de produire des estimations approximativement sans biais.

Non-reponse- partielle : on peut appliquer des méthodes de repondération mais le principale inconvénient est qu'il faut criées un nouveau poids ajuste pour chaque variable d'intérêt.

Traitement D :

- Imputation par la moyenne : on remplace chacune des valeurs manquantes par la valeur moyenne de l'ensemble de réponses obtenues.

- Imputation par le ratio : chaque valeur manquante yi est remplacée par la valeur prévuey_i^* obtenue par régression de y sur$x$.

- Imputation par régression : c'est une extension naturelle de l'imputation par la méthode du ratio ou l'on se sert de q variables auxiliaires $x_1, .., x_q$.

Méthode d'imputation	Moyenne	Ratio	Régression
Valeur imputée	$y_i^* = \frac{1}{r}\sum y_i = y_r$	$y_i^* = \frac{y_r}{x_r}$	$y_i^* = \beta_0 + \beta_1 x_{1i} + \beta_{qi}$

TABLE I.1 – méthode d'imputation

III. Analyse en composantes principales :

Lebart[2]

L'analyse en composante principales (ACP) fait partie du groupe des méthodes descriptives multidimensionnelles appelées méthode factorielle. L'ACP propose, à partir d'un tableau rectangulaire de données comportant les valeurs de p variables quantitatives pour n unités (appelées aussi individus), des représentations géométriques de ces unités et de ces variables. Ces données peuvent être issues d'une procédure d'échantillonnages ou bien de l'observation d'une population toute entière. Tableau de données sont les mesures effectuées sur n unités $u_1, u_2, .., u_n$.les p variables quantitatives qui représentent ces mesures sont $v_1, .., v_ip$. Le tableau des données brutes à partir duquel on va faire l'analyse est noté x et à la forme suivante :

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & x_{n,p} \end{pmatrix}$$

On peut représenter chaque unité par le vecteur de ses mesures sur les p variables : $U_i = [x_{i,1}, ...x_{i,p}]$ ce qui donne :

$$U_i = \begin{pmatrix} x_{i,1} \\ x_{i,j} \\ x_{i,p} \end{pmatrix}$$

Alors U_i est un vecteur de R^2

De façon analogue, on peut représenter chaque variable par un vecteur de R^n dont les composantes sont les vecteurs de la variable pour n unités :

$$V_j = \begin{pmatrix} x_{1,j} \\ x_{i,j} \\ x_{n,j} \end{pmatrix}$$

Choix d'une distance :
Pour faire une représentation géométrique, il faut choisir une distance entre deux points de l'espace la distance utilisée par l'ACP dans l'espace ou sont

Choix de l'origine :
Pour l'ACP, on choisit de donner le même poids $\frac{1}{n}$ à tous les individus Le centre de gravité G du nuage des individus est alors le point dont les coordonnées sont les valeurs moyennes des variables :

$$G = \begin{pmatrix} x_{i,1} \\ x_{i,j} \\ x_{n,j} \end{pmatrix}$$

Inertie total du nuage des individus :

On note IG le moment d'inertie du nuage des individus par rapport au centre de gravité G : formel

$$G = \begin{pmatrix} \frac{1}{N}\sum_{\hat{\imath}=1}^{n} x_{i,1} \\ \frac{1}{N}\sum_{\hat{\imath}=1}^{n} x_{i,j} \\ \frac{1}{N}\sum_{\hat{\imath}=1}^{n} x_{n,j} \end{pmatrix} = \begin{pmatrix} x_{.,1} \\ x_{.,j} \\ x_{.,p} \end{pmatrix}$$

Ce moment d'inertie total est intéressant car c'est une mesure de la dispersion du nuage des individus par rapport à son centre de gravité

$I_G = \frac{1}{N}\sum_{\hat{\imath}=1}^{n} d^2(G, u_i) = \frac{1}{N}\sum_{\hat{\imath}=1}^{n}\sum_{j=1}^{p}(x_{ij} - x_{.,j})^2$

$$= \frac{1}{N}\sum_{\hat{\imath}=1}^{n} U_{ci}^t U_{ci}$$

1. Interprétation des résultats :

1.1. Choix du nombre d'axes :

Le nombre d'axes à retenir est un problème délicat et qui n'a pas de solution rigoureuse.il faut tout d'abord éliminer les critères liés à une valeur a priori du pourcentage d'inertie expliquée : on trouve trop souvent des affirmations du genre « il faut 80% d'inertie expliquée « qui n'ont aucun sens.

S'il est souhaitable d'avoir beaucoup d'inertie expliquée sur un sous -espace cette valeur doit tenir compte du nombre de variables. Avoir 50% d ?inertie expliquée sur deux axes n'a pas le même sens avec 5 ou 50 variable au départ. Les seuls critères utilisables sont des critères empirique : celui de Kaiser est un des plus connus qui consiste à ne retenir que les composantes associées à des valeurs propres supérieures à un. En effet en données centrées- réduites les variables de départ ont des variances maximales donc supérieures.

Le critère du « coude » ou de Cattell consiste à détecter un ralentissement dans la décroissance des valeurs propres : au delà, si les valeurs propres sont peu différentes entre elles, il n'y a plus que du bruit.
Enfin il faut obtenir des composantes interprétables : ce n'est pas seulement une question d'habiller ni d'aptitude à l'expression verbale, on peut le formaliser en exigeant des corrélations suffisantes avec des variables supplémentaire.

2. Qualité de représentations :

Les projections sur les plans principaux sont des représentations déformées de la réalité et il convient de prendre des précautions.
Un usage bien établi consiste à se servir des cosinus carrées entre les projections et les points initiaux .des cosinus proches de 1 indiquent une bonne qualité de représentation .cependant des cosinus proches de 0 ne sont pas toujours les témoins d ?une mauvaise projection. forme

$$cos^2(a_i, k) = \frac{\langle \overrightarrow{G}_{u_i}, \overrightarrow{G}_a k \rangle^2}{\| \overrightarrow{G}_{u_i} \|^2}$$

3. Corrélation entre les variables :

Les composantes de l'ACP sont de nouvelles variables. Pour les interpréter en fonction des anciennes variables on calcule les coefficients de corrélation

$$R(\vec{c_k}; x_j)$$

Ces coefficients sont les coordonnées des variables initiales dans l'espace définie par les composantes principales.

Les cercles de corrélation sont les éléments essentiels pour l'interprétation « interne » des résultats .on observe ce que l'on appelle un effet « taille » lorsque toutes les variables initiales sont corrélées positivement entre elles.

Définition I.1 *Pour un modèle univariédichotomique, la matrice d'information de Fisher s'écrit sous la forme :*

$$I(\beta) = -E\left[\frac{\partial^2 logL(y,\beta)}{\partial\beta\partial\beta'}\right] = \sum_{i=1}^{N} \frac{f(x_i\beta)^2}{F(x_i\beta)[1 - F(x_i\beta)]} x_i^{'} x_i$$

CHAPITRE II

RÉGRESSION LOGISTIQUE

La régression logistique s'intéresse plus particulièrement à la description ou l'explication d'observations constituées d'effectifs comme, par exemple, le nombre de succès d'une variable de Bernoulli lors d'une séquence d'essais. Contrairement aux modèles de régression basés sur l'hypothèse de normalité des observations, les lois concernées sont discrètes et associées à des dénombrements : binomiale, multinationale.il s'agit de modéliser l'effet d'un vecteur de variables $x_1, .., x_k$ sur une variable aléatoire binomiale génériquement notée Y.

La régression logistique est un cas particulier du modèle linéaire généralisé. La régression logistique ou modèle logit est un modèle de régression binomiale. Comme pour tous les modèles de régression binomiale, il s'agit de modéliser l'effet d'un vecteur de variables $x_1, ..., x_K$ sur une variable aléatoire binomiale génériquement notée Y.

La régression logistique est un cas particulier du modèle linéaire généralisé.

Ce premier paragraphe présente certaines méthodes de régression sur variables qualitatives. Les méthodes d'inférence traditionnelles ne permettent pas de modéliser et d'étudier des caractères qualitatifs : des méthodes spécifiques doivent être utilisées tenant compte par exemple de l'absence de continuité des variables traitées ou de l'absence d'ordre naturel entre les modalités que peut prendre le caractère qualitatif.[4]

Le but de la plupart des recherches est de déterminer des relations entre un ensemble de variable. Les techniques « multivariables » ont été développées à cette fin. Souvent on considère une variable dépendante que l'on veut prédire et des variables indépendantes ou explicatives.

I. Modelés dichotomiques

Dans ce modèle, la variable expliquée ne peut prendre que deux modalités (variable dichotomique). On considère un échantillon de N individus. On pose, $\forall i \in [1, N]$:

$y_i = 1$ si l'événement s'est réalisé pour l'individu i
$y_i = 0$ si l'événement ne s'est pas réalisé pour l'individu i On a

$$\mathbb{E}(y_i) = \mathbb{P}(y_i = 1) \times 1 + \mathbb{P}(y_i = 0) \times 0 = \mathbb{P}(y_i = 1) = p_i$$

L'objectif des modèles dichotomiques consiste alors à expliquer la survenue de l'événement considéré en fonction d'un certain nombre de caractéristiques observées pour les individus de l'échantillon. On cherche donc des modèles qui consistent à spécifier la probabilité d'apparition de cet événement.[5]

1. Modèles Logit et Probit

Les modèles dichotomiques, Probit et Logit, admettent pour variable expliquée, non pas un codage quantitatif associé à la réalisation d'un événement (comme dans le cas de la spécification linéaire), mais la probabilité d'apparition de cet événement, conditionnellement aux variables exogènes. Ainsi, on considère le modèle suivant :[3]

$$p_i = \mathbb{P}(y_i = 1|x_i) = F(x_i\beta)$$

où la fonction F(.) désigne une fonction de répartition. Le choix de la fonction de répartition F (.) est a priori non contraint.

Toutefois, on utilise généralement deux types de fonction : la fonction de répartition de la loi logistique et la fonction de répartition de la loi normale centrée réduite. A chacune de ces fonctions correspond un nom attribué au modèle ainsi obtenu : modèle Logit et modèle Probit.

1.1. - Estimation des paramètres par maximum de vraisemblance

On cherche à estimer les composantes du vecteur β. La méthode la plus usitée, lorsque la loi des perturbations est connue, est la méthode du maximum de vraisemblance. Ceci permet de considérer les valeurs observées yi comme les réalisations d'un processus binomial avec une probabilité $F(x_i\beta)$. La vraisemblance des échantillons associés aux modèles dichotomiques s'écrit donc comme la vraisemblance d'échantillons associés à des modèles binomiaux. La seule particularité étant que les probabilités pi varient avec l'individu puisqu'elles dépendent des caractéristiques x_i. Ainsi la vraisemblance associée à l'observation y_i s'écrit sous la forme :

$$L(y_i, \beta) = p_i^{y_i}(1 - p_i)^{1-y_i}$$

$\forall x_i\beta \in \mathbb{R}$, on déduit alors la log-vraisemblance comme suit :

$$\log L(y\beta) = \sum y_i \log[F(x_i\beta)] + (1 - y_i) \log[1 - F(x_i\beta)]$$

L'estimateur du maximum de vraisemblance des paramètres β est obtenu en fonction de logvraisemblance $\log L(y\beta)$. En dérivant la log-vraisemblance par rapport aux éléments du vecteur β, de dimension $(K, 1)$, on obtient un vecteur de dérivées, note $G(\beta)$, appelé vecteur du gradient.

$$G(\beta) = \frac{\partial \log L(y\beta)}{\partial \beta}$$

Dans le cas du modèle Logit

$$G(\beta) = \sum (y_i - \Lambda(x_i\beta))\ x_i' \tag{II. 1}$$

où $\Lambda(x_i\beta)) = \dfrac{exp(x_i\beta)}{1 + exp(x_i\beta)}$ est la fonction de répartition de la loi logistique.

Remarque : Le système défini par l'équation II. 1 est non linéaire. L'estimateur de β ne peut être obtenu directement. Un algorithme d'optimisation numérique de la vraisemblance est donc nécessaire. Ces algorithmes se fondent à la fois sur le gradient mais aussi sur la matrice hessienne des dérivées secondes $H(\beta) = \frac{\partial^2 \log L(y\beta)}{\partial\beta\partial\beta'}$

2. Quelques définitions :

[4]

si l'on note $p_i = \mathbb{P}(y_i) = \Lambda(x_i\beta))$, étant donnée la définition de la loi logistique on remarque que plusieurs égalités, permettant de simplifier les calculs, peuvent être établies comme suit :

$$e^{x_i\beta} = p_i(1 + e^{x_i\beta}) \implies e^{x_i\beta} = \frac{p_i}{1 - p_i}$$

$\ln\left(\dfrac{p_i}{1 - p_i}\right) = x_i\beta$ est une fonction linéaire, et est appelé le logit de p, noté logit(p).

Définition II.1 *La fonction K(x)=logit(p(x)) est appelée une link function dans la théorie des modèles linéaires généralisés. On observe qu'elle peut varier entre $-\infty$ et $+\infty$.*

Remarque : l'interprétation des coefficients eux-mêmes demande plus de prudence : l'influence d'une variable sur la probabilité d'apparition étudiée p n'est en effet pas linéaire ; dans le cas du modèle logistique par exemple, elle l'est seulement sur son logit : $\ln\left(\dfrac{p}{1 - p}\right)$

Définition II.2 *la quantité $c_i = \dfrac{p_i}{1 - p_i}$ représente le rapport de la probabilité associée à l'événement $y_i = 1$ à la probabilité de non survenue de cet événement : il s'agit de la cote ("odds").*

Dans un modèle Logit, cette cote correspond simplement à la quantité $e^{x_i\beta}$

$$c_i = \frac{p_i}{1 - p_i} = e^{x_i\beta}$$

Si ce rapport est égal à ci pour l'individu i, cela signifie qu'il y a ci fois plus de chance que l'événement associé au code $y_i = 1$ se réalise, qu'il ne se réalise pas (" c_i contre 1" dans le langage usuel).

3. La constante du modèle

:

La constante du modèle s'interprète comme « l'effet de la catégorie de référence ». Autrement ditβ_0 permet de calculer la probabilité de y lorsque toute les co-variables $x_1, x_2, \ldots\ldots\ldots\ldots.., x_p$ sont nulles.

4. Les tests :

Les différentes méthodes d'estimation présentées précédemment conduisent à des estimateurs asymptotiquement normaux lorsque le nombre d'observations tend vers l'infini. Il est donc facile d'utiliser ces divers estimateurs pour construire des procédures de tests.

4.1. Test de Wald :

On considère le test
$H_0 : \beta_j = 0 \quad \text{contre} \quad H_1 : \beta_j \neq 0$
où β_j désigne la j^{ieme} composante du vecteur de paramètres $\beta \in \mathbb{R}^k$ d'un modèle dichotomique.

L'idée du test de Wald est d'accepter l'hypothèse nulle si l'estimateur $\hat{\beta}_j$ de β_jj est proche de 0.

L'estimateur $\hat{\beta}_j$ est asymptotiquement normal, avec l'écart-type estimé de l'estimateur $\sqrt{\hat{v}_{jj}}$ On définit la statistique de test :

$$W = \frac{(\hat{\beta}_j)^2}{\hat{v}_{jj}}$$

W suit la loi du chi-deux à un degré de liberté. On rejette alors l'hypothèse nulle avec un risque de première espèce σ lorsque la valeur de la statistique de test est supérieure au quantile d'ordre de la loi du chi-deux.

Puisque ω suit une loi du chi-deux asymptotiquement, on peut aussi considérer sa racine carrée

$$\zeta = \frac{\hat{\beta}_j}{\sqrt{\hat{v}_{jj}}}$$

qui suit asymptotiquement une loi normale.

5. Spécification linéaire des variables endogènes dichotomiques

Supposons que l'on dispose de N observations y_i, $i = 1, .., N$ d'une variable endogène dichotomique codée $y_i = 1$ ou $y_i = 0$, lorsque les observations de K variables exogènes sont $x_i = (x_i^1 ... x_i^K)$, $\forall i = 1, .., N$. Dans ce cas, le modèle linéaire simple s'écrit :

$$y_i = x_i\beta + \varepsilon_i \tag{II. 2}$$

où $\beta = (\beta_1, ..., \beta_K)' \in R^K$ d'ésigne un vecteur de K paramètres inconnus et où les perturbations ε_i sont supposées être indépendamment distribuées. On peut alors mettre en évidence plusieurs problèmes liés à l'utilisation de cette spécification linéaire simple pour modéliser une variable dichotomique.

On ne peut pas utiliser la même méthode que dans le cas continu puisqu'en particulier, la variable expliquée Y ne prenant que deux valeurs, la perturbation

suivrait obligatoirement une loi discrète, ce qui est incompatible avec les hypothèses habituelles de continuité et de normalité des résidus.

Pour toutes ces différentes raisons, la spéification linéaire des variables endogènes qualitatives, et plus spécialement dichotomiques, n'est jamais utilisée et l'on a recourt à des modèles logit ou probit

Définition II.3 *Dans le cas du modèle logit, la fonction de répartition $F(.)$ correspond à la fonction logistique* $\forall w \in R : F(w) = \frac{e^w}{1+e^w} = \frac{1}{1+e^{-w}} = \Lambda(w)$

Dans le cas du modéle probit, la fonction de répartition $F(.)$ correspond à la fonction de répartition de la loi normale centrée réduite $\forall w \in R : F(w) = \int_{-\infty}^{w} \frac{1}{\sqrt{2\pi}} e^{\frac{-z^2}{2}} dz = \Phi(w)$

6. Estimation des paramètres

On cherche à estimer les composantes du vecteur β. La méthode la plus usitée, lorsque la loi des perturbations est connue, est la méthode du maximum de vraisemblance.

6.1. Estimation par maximum de vraisemblance

[3]

Dans le cas du modèle dichotomique univarié, la construction de la vraisemblance est extrêmement simple. En effet, à l'événement $y_i = 1$ est associé la probabilité $p_i = F(x_i\beta)$ et à l'événement $y_i = 0$ correspond la probabilité $1 - p_i = 1 - F(x_i\beta)$.

Ceci permet de considérer les valeurs observées y_i comme les réalisations d'un processus binomial avec une probabilité $F(x_i\beta)$. La vraisemblance des échantillons associés aux modéles dichotomiques s'écrit donc comme la vraisemblance d'échantillons associés à des modèles binomiaux. La seule particularité étant que les probabilités p_i varient avec l'individu puisqu'elles dépendent des caractéristiques x_i. Ainsi la vraisemblance associée à l'observation y_i s'écrit sous la forme :

$$L(y_i, \beta) = p_i^{y_i}(1 - p_i)^{1-y_i}$$

$\forall x_i\beta \in R$, on déduit alors la log-vraisemblance comme suit :

$$logL(y, \beta) = \sum_{i=1}^{N} y_i log[F(x_i\beta)] + (1 - y_i)log[1 - F(x_i\beta)]$$

L'estimateur du maximum de vraisemblance des paramètres β est obtenu en maximisant soit la fonction de vraisemblance $L(y, \beta)$ soit la fonction de log-vraisemblance $logL(y, \beta)$. En dérivant la log-vraisemblance par rapport aux éléments du vecteur β, de dimension $(K, 1)$, on obtient un vecteur de dérivées, note $G(\beta)$, appelé vecteur du gradient.

$$G(\beta) = \frac{\partial logL(y, \beta)}{\partial \beta} = \sum_{i=1}^{N} y_i \frac{f(x_i\beta)}{F(x_i\beta)} x_i^{'} + (y_i - 1)\frac{f(x_i\beta)}{1 - F(x_i\beta)} x_i^{'}$$

où $f(.)$ est la fonction de densitéassociée à $F(.)$ et où x_i' désigne la transposée du vecteur x_i de dimension $(1, K)$. Dans le cas du modèle logit, ce système se raméne à :

$$G_L(\widehat{\beta}) = \sum_{i=1}^{N}(y_i - \Lambda(x_i\widehat{\beta}))x_i^{'} = 0$$

Dans le cas du modéle probit, on a :

$$G_P(\widehat{\beta}) = \sum_{i=1}^{N} \frac{\left(y_i - \Phi(x_i\widehat{\beta})\right)\Phi(x_i\widehat{\beta})}{\Phi(x_i\widehat{\beta})\left[1 - \Phi(x_i\widehat{\beta})\right]}x_i^{'} = 0$$

6.2. Propriétés asymptotiques des estimateurs

Rappelons que l'estimateur $\widehat{\beta}$ du maximum de vraisemblance du vecteur de paramètre $\beta \in R^K$ dans un modèle dichotomique est défini par la résolution du système de K équations non linéaires en β :

$$\frac{\partial logL(y,\widehat{\beta})}{\partial\widehat{\beta}} = \sum_{i=1}^{N} \frac{[y_i - F(x_i\widehat{\beta})]f(x_i\widehat{\beta})}{F(x_i\widehat{\beta})[1 - F(x_i\widehat{\beta})]}x_i^{'} = G(\widehat{\beta}) = 0 \qquad \text{(II. 3)}$$

Remarque :

Le système défini par l'équation (II. 3) est non linéaire. L'estimateur $\widehat{\beta}$ ne peut être obtenu directement. Un algorithme d'optimisation numérique de la vraisemblance est donc néecessaire. Ces algorithmes se fondent à la fois sur le gradient mais aussi sur la matrice hessienne des derivéees secondes $H(\beta) = \frac{\partial^2 logL(y,\beta)}{\partial\beta\partial\beta'}$. C'est pourquoi, nous allons donner l'expression des gradients et des matrice hessiennes, dans le cas particulier des modèles logit et probit.

Pour un modèle dichotomique univarié, la matrice hessienne associée à la log vraisemblance d'un échantillon de taille N, noté $y = (y_1, .., y_N)$, s'écrit sous la forme :

$$\begin{aligned} H(\beta) &= \frac{\partial^2 logL(y,\beta)}{\partial\beta\partial\beta'} \\ &= -\sum_{i=1}^{N}\left[\frac{y_i}{F(x_i\beta)^2} + \frac{1-y_i}{[1-F(x_i\beta)]^2}\right]f(x_i\beta)^2 x_i^{'}x_i + \\ &\quad \sum_{i=1}^{N}\left[\frac{y_i - F(x_i\beta)}{F(x_i\beta)[1-F(x_i\beta)]}\right]f^{'}(x_i\beta)x_i^{'}x_i \end{aligned}$$

où $f^{'}(.)$ désigne la dérivée de la fonction de densité $f(.)$ associée à $F(.)$. Il n'existe pas d'expression simplifiée dans le cas des modèles logit et probit de la matrice hessienne.

L'espérance de la matrice hessienne est donné par :

$$E[H(\beta)] = -\sum_{i=1}^{N}\frac{f(x_i\beta)^2}{F(x_i\beta)[1-F(x_i\beta)]}x_i^{'}x_i$$

On reconnaît ici bien sûr, l'expression de l'opposé de la matrice d'information de Fischer $I(\beta)$.

Définition II.4 *Pour un modèle univarié dichotomique, la matrice d'information de Fisher s'écrit sous la forme :*
$I(\beta) = -E\left[\frac{\partial^2 logL(y,\beta)}{\partial\beta\partial\beta'}\right] = \sum_{i=1}^{N} \frac{f(x_i\beta)^2}{F(x_i\beta)[1-F(x_i\beta)]} x_i' x_i$

Dans le cas du modèle logit, cette matrice est définie par :

$$I(\beta) = \sum_{i=1}^{N} \lambda(x_i\beta) x_i' x_i$$

Dans le cas du modèle progit, cette matrice est définie par :

$$I(\beta) = \sum_{i=1}^{N} \frac{\phi(x_i\beta)^2}{\Phi(x_i\beta)[1-\Phi(x_i\beta)]} x_i' x_i$$

6.3. L'algorithme de maximisation de la vraisemblance

Toutes les fois que la fonction de logvraisemblance est globalement concave, comme pour les modèles logit et probit, il existe de nombreuses manières différentes d'estimer facilement les modèles à réponse binaire. Une approche qui fonctionne généralement bien consiste à utiliser un algorithme.

La procédure employée dans la plupart des cas, est basée sur l'algorithme de Newton-Raphson. On utilise souvent une autre procédure, celle de l'algorithme de Fisher (Fisher scoring). Cet algorithme ressemble à celui de Newton-Raphson, la différence étant que le Fisher scoring utilise l'espérance de la matrice des dérivées secondes au lieu de la matrice elle-même.

-Dans la méthode de Newton-Raphson, on a :

$$\beta^{(k+1)} = \beta^{(k)} - (H^{(k)})^{-1} q^{(k)}$$

où H est la matrice hessienne, q est le vecteur des dérivées. $H^{(k)}$ et $q^{(k)}$ sont évaluées en $\beta = \beta^{(k)}$.

-La formule du Fisher scoring s'écrit :

$$\beta^{(k+1)} = \beta^{(k)} - (I(\beta^{(k)}))^{-1} q^{(k)}$$

où $I(\beta^{(k)})$ est la $k^{ième}$ approximation de la matrice d'information de Fisher estimée.

On utilise ces méthodes itératives jusqu'à obtenir la stabilité, en l'occurrence jusqu'au moment où la valeur absolue de la différence entre les valeurs calculées pour le logarithme de la vraisemblance à deux étapes successives soit en deçà d'un seuil fixé à l'avance. Toutefois, on considère que les itérations ont convergé lorsque la différence maximale entre les estimateurs des différents paramètres est inférieure à un seuil, par défaut 10^{-4}.

6.4. Lois et variance asymptotique de l'estimateur du maximum de vraisemblance :

Dans le cas d'un modèle dichotomique logit ou probit, l'estimateur $\widehat{\beta}$ du maximum de vraisemblance est défini par :

$$\widehat{\beta} = \arg\max\left(\tfrac{1}{n}\log L(y,\beta)\right)$$

proposition Sous certaines conditions, l'estimateur du maximum de vraisemblance est convergent et suit asymptotiquement une loi normale : $\sqrt{N}(\widehat{\beta} - \beta_0) \to^L N(0, I(\beta_0)^{-1})$ où est la vraie valeurdes paramètres.

II. Modelés Polytomiques :

La régression logistique adaptée à la modélisation d ?une variable dichotomique se généralise au cas d ?une variable Y à plusieurs modalités ou polytomique. Si ces modalités sont ordonnées, on dit que la variable est qualitative ordinale. Ces types de modélisation sont très souvent utilisés en épidémiologie et permettent d ?évaluer ou comparer des risques par exemples sanitaires. Des estimations d ?odds ratio ou rapports de cotes sont ainsi utilisés pour évaluer et interpréter les facteurs de risques associés à différents types (régression polytomique) ou seuils de gravité (régression ordinale) d ?une maladie.

Il est de coutume d'appliquer les modèles logit de choix binaire (ou encore modèles dichotomiques) dès que la variable à expliquer ne peut prendre deux modalités. Mais dans la pratique, une variable qualitative peut prendre aussi plusieurs modalités comme par exemple : le choix entre autant de candidats lors de la présidentielle. Alors dans ce cas, les modèles à choix multiple sont exigés.

Les modèles à choix multiple sont une généralisation des modèles binaires. Dans ces modèles la variable à expliquer, qualitative, n'est donc plus binaire (0 et 1), mais polytomique (ou multinomiale). Nous différencions, en fonction du type de la variable à expliquer les modèles ordonnés et les modèles non ordonnés.

Dans ces modèles, les valeurs des coefficients des modèles ne sont pas directement interprétables en terme de propension marginale, seuls les signes des coefficients indiquent si la variable agit positivement ou négativement sur la variable latente.

Les résultats d'estimation s'apprécient de la même manière que pour les modèles de choix binaire.

Ce sont des modèles dans lesquels la variable expliquée peut prendre plus de deux modalités.

On considère un modèle multinomial dans le lequel la variable dépendante qualitative observée pour le jieme individu $\forall i = 1, ..., N$ notée y_i , peut prendre $m + 1$ modalités indicées $j = 0, 1, 2, .., m$ supposées mutuellement exclusives pour chaque individu i.
La probabilité associée à chaque réponse est définie par :

$$\mathbb{P}(y_i = j) = F_{ij}(x\beta) \qquad \forall i = 1, ..., N \ , \ \forall j = 0, 1, ..., m$$

où la fonction de répartition $F_{ij}(x\beta)$ correspond à la probabilité que l'individu i choisisse la modalité j en fonction des variables explicatives x et du vecteur de paramètres β. Dans un modèle multinomial, la probabilité associée à la $(m + 1)^{eme}$ modalité (généralement l'événement codé en 0) n'a pas besoin d'être spécifiée puisqu'elle peut être calculée à partir des m probabilité comme suit :

$$\mathbb{P}(y_i = j) = \sum_j F_{ij}(x\beta) \qquad \forall i = 1, ..., N$$

Si l'on définit les variables binaires y_{ij} telles que :

$y_{ij} = 1$ si $y_i = j$

$y_{ij} = 0$ sinon

$\forall i = 1, ..., N$, $\forall j = 0, 1, ..., m$
alors on peut écrire la vraisemblance associé à l'échantillon
$y = (y_{10}, ..., y_{1m}, ..., y_{N0}, ..., y_{Nm})$ comme le produit des probabilités associées aux différentes modalités, et ceci pour tous les individus :

$$L(y, \beta) = \prod_i \prod_j F_{ij}(x\beta)^{y_{ij}}$$

Les résultats généraux concernant les estimateurs du MV et les propriétés asymptotiques des estimateurs étudiées dans le paragraphe 1 concernant les modèles binaires restent valables ici. Il n'y a donc pas de difficulté technique concernant l'estimation des paramètres de ces modèles.

Les modèles ordonnés sont utilisés lorsque les valeurs prises par la variable multinomiale correspondent à des intervalles dans lesquels va se trouver une seule variable latente inobservable continue. Ainsi, un modèle polytomique univarié ordonné est un modèle dans lequel on a une variable, plusieurs modalités, et un ordre naturel sur ces modalités.

définition

Définition II.5 *Un modèle polytomique univarié ordonné peut s'écrire sous la forme :*

$$\begin{array}{rcl} y_i = 0 & si & y_i^* < c_1 \\ y_i = 1 & si & c_1 \leq y_i^* < c_2 \\ y_i = m & si & y_i^* > c_m \\ \forall \quad i = 1, ..., N & & \end{array}$$

avec $c_{j+1} \geq c_j$ et où la variable latente y_i^* est défini par $y_i^* = x_i\beta + \epsilon_i$

avec $x_i = (x_i^1..x_i^K)$, $\forall i = 1, .., N$, $\beta = (\beta_1, ..., \beta_K) \in \mathbb{R}^K$, ϵ_i i i.i.d.$(0, \sigma_{\epsilon^2})$ et où $\epsilon_i/\sigma_\epsilon$ suit une loi de fonction de répartition $F(.)$.
Si la fonction $F(.)$ correspond à la loi logistique, $F(.) = \Lambda(.)$, le modèle est un modèle logit multinomial ordonné.
A partir de la définition précédente on peut déduire la loi de la variable qualitative observée y_i qui nous servira par la suite à construire la fonction de vraisemblance.
En effet, on a :

$$\mathbb{P}(y_i = 0) = \mathbb{P}(y_i^* < c_1) = F(c_1/\sigma_\epsilon - x_i\beta/\sigma_\epsilon)$$

$$\mathbb{P}(y_i = 1) = \mathbb{P}(c_1 \leq y_i^* < c_2) = F(c_2/\sigma_\epsilon - x_i\beta/\sigma_\epsilon) - F(c_1/\sigma_\epsilon - x_i\beta/\sigma_\epsilon)$$

$$\mathbb{P}(y_i = m) = \mathbb{P}(y_i^* > c_m) = 1 - F(c_m/\sigma_\epsilon - x_i\beta/\sigma_\epsilon)$$

De façon générale, on obtient le résultat suivant :
Dans un modèle polytomique univarié ordonné satisfaisant la définition, la probabilité associée à l'événement $y_i = j$, $\forall j = 0, 1, .., m$ est définie par :
$\mathbb{P}(y_i = j) = F(c_{j+1}/\sigma_\epsilon - x_i\beta/\sigma_\epsilon) - F(c_j/\sigma_\epsilon - x_i\beta/\sigma_\epsilon) \quad \forall i = 1, ..., N$
avec par convention $c_0 = -\infty$ et $c_{m+1} = \infty$

Il ne reste plus alors qu'à construire la vraisemblance associée à l'échantillon y comme suit :

$$L(y, \beta, c_1, ..., c_m, \sigma_\epsilon) = \prod_i \prod_j \mathbb{P}(y_i = j)^{y_{ij}} = \prod_i \prod_j \Big[F(c_{j+1}/\sigma_\epsilon - x_i\beta/\sigma_\epsilon) - F(c_j/\sigma_\epsilon - x_i\beta/\sigma_\epsilon)\Big]^{y_{ij}}$$

où la variable dichotomique yij est définie par :
$y_{ij} = 1$ si $y_i = j$
$y_{ij} = 0$ sinon

$\forall i = 1, ..., N \quad \forall j = 1, ..., m$

$$\implies \log L(y, \beta, c_1, ..., c_m, \sigma_\epsilon) = \sum_i \sum_j y_{ij} \log \Big[F(c_{j+1}/\sigma_\epsilon - x_i\beta/\sigma_\epsilon) - F(c_j/\sigma_\epsilon - x_i\beta/\sigma_\epsilon)\Big]$$

où $F(.)$ est une fonction de répartition donnée.

Logit multinomial indépendant

Le modèle logit multinomial est obtenu lorsque les paramètres β_j diffèrent selon les modalités et que les variables explicatives varient uniquement en fonction des individus. Dès lors, on peut définir la forme générale de la probabilité que l'individu i choisisse la modalité j de la façon suivante :

Dans un modèle logit multinomial, la probabilité que l'individu i choisisse la modalité j, $\forall j = 1, ..., m$, est définie par :

$$\mathbb{P}(y_i = j) = \exp(x_i\beta_j) / \sum_k \exp(x_i\beta_k)$$

où le vecteur β_0 est normalisé à zéro $\beta_0 = 0$. Sous l'hypothèse de normalisation $\beta_0 = 0$, la probabilité associée à la modalité de référence 0 est définie par :

$$\mathbb{P}(y_i = 0) = 1 / \sum_k \exp(x_i\beta_k)$$

La log vraisemblance associée à un modèle logit multinomial à $m + 1$ modalités $j = 0, 1, .., m$ s'écrit :

$$\log L(y, \beta_1, \beta_2, ..., \beta_m) = \sum_i \sum_j y_{ij} x_i\beta - \sum_i \log[1 + \sum_k \exp(x_i\beta_k)]$$

avec $\beta_0 = 0$ par convention.

CHAPITRE III

MÉTHODE DE DISCRIMINATION BASÉE SUR LA CONSTRUCTION D'UN ARBRE DE DÉCISION BINAIRE

Les travaux de (Morgan et Sonquist) [23], ont été le point de départ pour les développements des techniques de segmentation ou de discrimination par arbre. Mais les premiers développements dans ce domaine on été lancer par [Hunt, Marin et Stone, 1966] [Messenger et Mandell, 1972], les modifications introduites par Quinlan en 1979, 1983 et 1986 et l'approche de [Breiman et al.][7] ont beaucoup contribué à la grande popularité de cette technique, et celle des arbres de décision. En fait, les arbres de décisions n'ont pas surgi par hasard ou d'une façon abstraire, ils sont apparus pour résoudre des problèmes complexes, qui ne pouvaient pas être résolus par les autres méthodes de discrimination existantes.

I. Théorie des graphes :

Introduction

L'histoire de la théorie des graphes débute peut-être avec les travaux d'Euler au XVIIIe siècle et trouve son origine dans l'étude de certains problèmes, tels que celui des ponts de Kônigsberg. La théorie des graphes s'est alors développée dans diverses disciplines telles que la chimie, la biologie, les sciences sociales. Depuis le début du XXe siècle, elle constitue une branche à part entière des mathématiques, grâce aux travaux de Kônig, Menger, Cayley puis de Berge et d'Erdôs. De manière générale, un graphe permet de représenter la structure, les connexions d'un ensemble complexe en exprimant les relations entre ses éléments : réseau de communication, réseaux routiers, interaction de diverses espèces animales, circuits électriques.
Les graphes constituent donc une méthode de pensée qui permet de modéliser une grande variété de problèmes en se ramenant à l'étude de sommets et d'arcs. Les derniers travaux en théorie des graphes sont souvent effectués par des informaticiens, du fait de l'importance qu'y revêt l'aspect algorithmique.

1. Généralité sur les graphes :

[6]

Soit$G = [X, U]$ un graphe dont l'ensemble des sommets est X et U l'ensemble de ses arêtes.

Définition III.1 *Une chaîne de longueur q est une séquence de q arêtes.*

$$L = \{u_i, u_2, ..., u_n\}$$

Telle que chaque arcs u_r de la séquence $(2 < r < q - l)$ ait une extrémité commune avec l'arc $u_{r-1}(u_{r-1} \neq u_r)$ et l'autre extrémité commune avec l'arc $u_{r+1}(u_{r+i} \neq u_r)$. Une chaîne qui n'utilise pas deux fois la même arête est dite simple.

Définition III.2 *Un cycle est une chaîne simple*

Définition III.3 *un graphe connexe : c'est un graphe tel que pour toute paire x, y de deux sommets distincts, il existe une chaîne reliant ces deux points*

Définition III.4 *Un arbre est un graphe connexe sans cycles.*

Définition III.5 *un sommet pendant est un sommet qui n'est adjacent qua un seul sommet.*

Définition III.6 *dans un graphe on appelle racine un point (sommet) a tel que tout autre sommet du graphe puisse être atteint par un chemin issue de a .*
Un arbre de décision est un graphe ou un diagramme représente un système de classification ou un modèle de prédiction.

Théorème 1 *Soit $G = (X, U)$ un graphe d'ordre $|X| = n > 2$, les propriétés suivantes sont équivalentes pour caractériser un arbre :*

a) G est un connexe et sans cycles.

b) G est sans cycle et admet $n - 1$ arêtes.

c) G est connexe et admet $n - 1$ arêtes.

d) G est sans cycles et on ajoutant une arête, on crée un cycle (et un seul).

e) G est connexe, et si on supprime une arête quelconques, il n'est plus connexe.

f) Tout couple de sommets est relié par une chaîne et une seule.

II. Arbre de décision binaire

La segmentation couvre un ensemble de méthode permettant la construction d'un graphe d'induction. Elle explique une variable qualitative ou quantitative à l'aide de tout type de critère. Elle cherche à résoudre les problèmes de discrimination et de régression en segmentant de façon progressive l'échantillon pour obtenir un arbre de décision.

Il existe différentes méthodes de segmentation, AID (Automatique Interaction Detector) CAID, CART (Classification And Régression Trees) ; SIPINA, on va s'intéresser à la méthode de CART.

Propriété

Les arbres permettent le traitement des cas où les variables sont nombreuses, grâce à leur sélection automatique.

Un autre aspect très important concerne les règles de décision résultante ; celles-ci sont très simples, d'utilisation d'interprétation et ont à la fois un pouvoir explicatif et décisionnel, ce qui les rend très recherchées par les praticiens. Les arbres de décision sont capables de résoudre des problèmes difficiles, même complexes.

1. construction des arbres de décision binaires

L'utilisation des arbres binaires remontre au programme 'A.I.D' (Automatic Interaction Détection) proposé par J.A. MORGAN et J.N. SONQUIST [23] dans les années 60. Les importants développements théoriques récents sont dû à L. BREIMAN et Al [23],[2]. qui proposent de construire un arbre binaire sans s'imposer de règle d'arrêt de la procédure de division des nœuds.
Dans la méthode présentée, les variables explicatives peuvent être de nature quelconque, Mais toute variable quantitative doit être préalablement transformée en une variable qualitative ayant suffisamment de modalités de sorte que la perte d'information entraînée par ce codage soit négligeable.

1.1. principe

Dans la construction d'un arbre, la seule connaissance a priori se réduit à une description de l'ensemble d'apprentissage. Les données sont constituées de l'observation de p variables qualitatives ou quantitatives explicatives X^j et d'une variable à expliquer y qualitative a m modalités $\{\zeta_I.I \doteq 1, ..., m\}$ ou quantitative réelle observée sur un échantillon de n individus.

La construction d'un arbre de discrimination binaire consiste à déterminer une séquence de nœud. L'idée de base pour la construction d'un arbre de décision binaire est d'effectuer la division d'un nœud, cette procédure nécessite de définir un critère permettant de sélectionner la meilleure division d'un nœud.

Dans notre travail on s'intéresse au cas de la discrimination car la variable y est nominale et répartie en K classes, la sélection d'une division doit être telle que les segments descendants soient plus purs que le nœud parent. Autrement dit, il faut que le mélange des classes soit moins important dans le segment descendant que dans le nœud parent.

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.

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.

A la racine ou nœud initial correspond l'ensemble de l'échantillon.

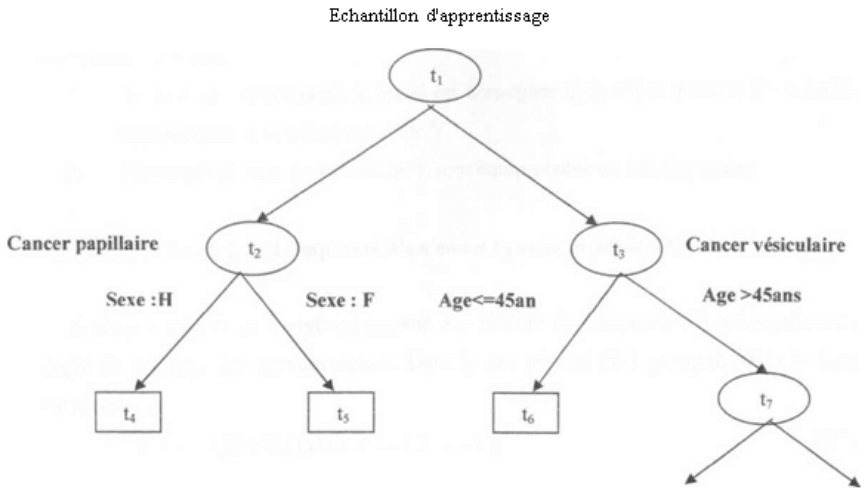

FIGURE III.1 – arbre de décision binaire

La figure III.1 représente un arbre de décision binaire illustratif où l'on distingue deux types de nœuds : Les nœuds intermédiaires (nœuds non terminaux) entourés d'un cercle : ce sont des nœuds qui fournissent deux descendants immédiats : par exemple t_3 qui se divise en t_6, t_7 ;

Les nœuds terminaux (feuilles de l'arbre) entourés d'un carré : ce sont les nœuds qui ne sont plus divisés.

1.2. 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 $2^{m-1} - l$.
Le critère de division repose sur la définition d'une fonction d'hétérogénéité ou de désordre. 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 :[19]

1- Nulle si et seulement si, le nœud est homogène c'est-à-dire que tout les individus appartiennent à la même valeur de Y.

2- Maximale lorsque les valeurs de Y sont équiprobable ou très dispersées.

1.3. Réduction de L'impureté d'un nœud t par la division

[6]

A chaque nœud t de l'arbre est associé une mesure de l'impureté $i(t)$ qui représente le degré de mélange des groupes dans t. Dans le cas général de k groupes, $i(t)$ a la forme suivante :

$$i(t) \doteq 2(\sum[P(\frac{r}{t}).P(\frac{s}{t}); r > s \doteq 1, 2, ..., k]$$

où $P(\frac{r}{t})$ est la proportion de sujets du groupe G_r dans le nœud t.
en utilisant le résultat :

$1 \doteq [\sum_r P(\frac{r}{t})]^2 \doteq \sum_r[P^2(\frac{r}{t})] + 2\sum_{rs} P(r/t).P(\frac{r}{t}); r > s]$ il vient :

$$i(t) = 1 - \sum_r[P^2\frac{r}{t}]; r = l, 2, ..., k$$

Un nœud est dit « pur » s'il ne contient que des sujets d'un seul groupe, dans ce cas :

$$i(t) = 0$$

Plus le mélange des groupes dans t est important, plus l'impureté $i(t)$ est élevée. Dans le cas particulier de deux groupes ((k = 2) :

$$i(t) = 2P(l/t).P(2/t)$$

chaque division d d'un nœud t parla variable x_j entraine une réduction de l'impureté l'expression est :

$$\Delta_j^m = i(t) - \lfloor p_g i(t_g) + p_d i(t_d) \rfloor$$

Où :tg est Le segment gauche qui contient les sujets vérifiant $X_j \epsilon m_1$ ettd le segment droite qui contient les sujets qui vérifient $X_j \epsilon m_2$. p_g et p_d sont les proportions des sujets de t allant respectivement dans les descendants t_g ett_d.

Δ_j^m Représente la différence entre l'impureté du nœud parent et la moyenne pondérée des impuretés de ses nœuds descendants immédiats.

1.4. Critère de la pureté maximale

[7]

$P(j/t)$ est la proportion de la classe j dans le segment t : c'est le nombre d'individus appartenant à la classe j dans le segment t. L'impureté $i(t)$d'un nœud t est une fonction non négative f de $p(l/t), ..., p(k/t)$ qui vérifie les conditions suivantes :

1- f est maximale pour $p(\frac{r}{t})$: l'impureté d'un nœud est maximale quand pour ce œud les probabilités d'appartenance aux différents groupes sont égales entre elles.

2- f est nulle pour $p(\frac{r}{a}) = 1$ et $p(\frac{s}{a}) = 0$ pour $r \neq s, (r = 1, ..., k; s = 1, ...k)$; l'impureté est nulle dés que le nœud t ne contient que des observations d'un seul groupe.

3- f est une fonction symétrique des probabilités $p(\frac{r}{a}), (r = 1, ..., k)$.

Propriétés :

Si la fonction f définissant l'impureté est strictement concave, alors toute division d'un nœ ud conduit à une réduction positive ou nulle de l'impureté.

$$\Delta_j^m \geqslant 0$$

La nullité de Δ_j^m étant obtenue si et seulement si l'égalité $p(\frac{r}{t_g}) = p(\frac{r}{t_d}) = p(\frac{r}{t})$ est vraie pour tout r variant de 1 à k.

$$\Delta_j^m = 0 \Leftrightarrow p(\frac{r}{t_g}) = p(\frac{r}{t_d}) = p(\frac{r}{t}) \forall r$$

En effet, la stricte concavité de la fonction f implique :

$$P_g I(t_g) + p_d I(t_d) = p_g f(p(1/t_g), ..., p(k/t_g) p_d f(p(1/t_d, ..., p(k/t_d))$$

$$\Longleftarrow f(p_g(p(1/t_g) + p_d(1/t_d), ..., P_g p(k/t_g) + p_d(p(k/t_d))$$

.

L'égalité se produisant si et seulement si :

$$p_d(p(r/t_d) = p_g(p(r/t_g)$$

pour tout r variant de 1 à k, La relation est vraie.

Par conséquent pour chaque variable X_j, la meilleure division d_j* est telle que la réduction de

l'impureté Δ_j^* est maximale :

$$\Delta_j^* = max \Delta_j^m$$

où d_j est l'ensemble des p variables, la division du nœud t est effectuée à l'aide de la variable qui assure :

$$\Delta^* = max \Delta_j^*$$

Choisir la division Δ^* sur le nœud t qui maximise la réduction de l'impureté de l'arbre est équivalent à choisir la division qui donne le maximale de l'impureté du nœud t.

1.5. critère d'arrêt

La croissance de l'arbre s'arrête à un n ?ud donné, qui devient 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 très faible (généralement inférieur à une valeur seuil comprise en général entre let 5).

2. algorithme générale de la segmentation

[19]
Les étapes de l'algorithme sont les suivantes :

1. au départ, on dispose d'un seul segment contenant l'ensemble des individus.
2. A la première étape, la procédure de construction de l'arbre examine une par une toutes les variables explicatives.

   Chaque division scinde l'échantillon en segments descendants :

   Le segment gauchet_g contient les sujets vérifiant $X_1 \in m_1$ et le segment droite contient les sujets qui vérifient $X_1 \in m_1$.

   La procédure sélectionne la meilleure division parmi toutes les variables explicatives, au sens d'un critère de division adopté. Ainsi pour chaque variable on obtient la meilleure division qui fournit les deux segments les plus purs vis à vis de y.
3. A l'étape suivante, on applique la même procédure à chacun des deux segments descendants obtenus.
4. La procédure s'arrête lorsque tous les segments sont déclarés terminaux, soit parce qu'ils ne nécessitent plus de division soit parce que leur taille est inférieure à un effectif fixé au préalable.
5. Pour un nouvel individu, on définit une règle d'affectation simple.

III. Méthode CART

La méthode CART (Classification And Régression Trees) est proposée par Breiman et Al (1984)[7], elle est le produit de dix années de travaux et de recherche, ce qui lui assure des performances stables, elle présente des avantages importants dont le premier est la stabilité des règles d'affectation, l'interprétation des résultats étant directe et intuitive.
Par ailleurs la méthode CART, contrairement aux autres méthodes de segmentation, n'impose aucune règle d'arrêt de division de segment. Elle fournie à partir de l'arbre binaire complet la séquence des sous arbres obtenues en utilisant une procédure d'élagage.

1. Principe général de la méthode CART

La méthode de discrimination par arbre binaire CART, proposée par Breiman, Friedman, Olshen et Stone (1984), inclut des solutions qui répondent aux critiques

les plus importants faîtes aux arbres de décision.
La construction d'un arbre de décision binaire s'effectue d'une façon récursive à travers une division successive de l'ensemble d'états qui correspond à la racine de l'arbre en sous-ensembles correspondants aux n ?uds descendants.

2. Les critères de segmentations

L'idée fondamentale dans la construction d'un arbre de décision binaire par la méthode CART est la sélection dans chaque nœud t, d'un attribut binaire de façon à ce que les nœuds descendants t_g et t_d soient purs que le nœud parent t.

La méthode CART comporte divers critères de choix de l'attribut binaire : Gini, Symgini, Twoing, les moindres carrés, Shannon, critère de kh_2,..ect.

2.1. indice de Gini

Le coefficient de Gini consiste à choisir la variable binaire$W$ qui coupe t en t_g,$t_d$qui maximise la diminution de l'impureté, donnée par :

$$f(p_1^t, p_2^t, ..., p_k^t) = \sum p_i^t p_j^t = 1 - \sum (p_i^t)^2$$

3. Le coût d'une erreur de classement est une fonction non symétrique sur les classes

[19]

4. procédure de sélection

Il est nécessaire de diviser l'échantillon de base en deux parties, l'échantillon d'apprentissage constitué par 2/3 de l'échantillon de base et l'échantillon de test le tiers restant.

La recherche du meilleur sous arbre G^* se fait de la façon suivante :

A partir de l'échantillon d'apprentissage, on construit l'arbre complet G_{max} ou chaque segment terminal contient un nombre restreint d'individus. Puis l'opération d'élagage de l'arbre G_{max} consiste à construire une séquence optimale de sous arbres emboîtés $G_H, ..., G_h, ..., G_1$, où G_H coïncide avec G_{max}, G_h est le sous arbre ayant $h$segment terminaux et G_1 est l'échantillon total.

Chaque sous arbre G_h de cette séquence est optimal au sens suivant : son erreur apparente est minimale parmi les sous arbres ayant le même nombre de segment terminaux, si S_h est l'ensemble des sous arbres de G_{max} ayant h segments terminaux alors :

$$\xi(G_k) = \min_{G \in S_h} \xi(G)$$

A partir de l'échantillon test, on sélectionne parmi les sous arbres de la séquence optimale, le meilleur sous arbre G^*, qui présente la plus petite erreur théorique :

$$\xi(G_*) = \min_{1 \leq h \leq H} \xi(G)$$

Les individus de l'échantillon test parcourent chacun des sous arbres de la séquence optimale et atterrissent dans un segment terminal, ce qui entraîne une estimation de l'erreur théorique pour le sous arbre.

5. L'élagage de l'arbre

Définition III.7 *L'élagage est l'étape qui consiste a supprimer les parties de l'arbre qui ne semblent pas performantes pour prédire la classe de nouveaux cas et remplacées la par un nœud terminal (associé à la classe majoritaire)*

5.1. Principe général

De façon générale, considérons le coût de mauvais classement $\zeta(t)$, qui correspond usuellement au taux d'erreur moyen calculé sur l'échantillon d'apprentissage. Soit G_{max} l'arbre le plus fin que nous puissions obtenir par l'algorithme de construction précédent. On notera S_{max} la partition terminale associée à G_{max} et t, $\zeta(S_{max})$le taux d'erreur correspondant.

$\zeta(S_{max})$ étant une estimation optimiste du taux d'erreur théorique ζ^* inconnu, généralement $\zeta(S_{max}) < \zeta^*$

Le principe de l'élagage par CART repose sur un critère « coût complexité », qui vise à pénaliser la prolifération de sommets.

5.2. Coût de complexité d'un arbre

Considérons un arbre binaire G_{max} dont la partition terminale est notée $t = t_1, ..., t_k$

Soit $\alpha \geq 0$; le coût de complexité de l'arbre G_{max} est définit par :

$$C_\alpha(G_max) = \zeta(t) + \alpha card(t)$$

or card(t)=K
Et
$\zeta(t) = \sum \zeta(t)$le taux d'erreur sur chaque sommet terminal
Par conséquent

$$C_\alpha(G_max) = \sum[\zeta(t) + \alpha] = \sum C_\alpha$$

Soit t un sommet non terminal de l'arbre G, le sous arbre descendant de t noté $G(t^+)$, est le sous arbre de G dont la racine est s.

6. Algorithme d'élagage

Considérons l'arbre de taille maximum G_{max}, on notera alors S l'ensemble des sommets, S_{ter} l'ensemble des sommets terminaux, S_{int} l'ensemble des sommets intermédiaire.

$$S \doteq S_{ter} \cup S_{int}$$

L'algorithme de l'élagage se déroule en deux étapes :

La première consiste à construire une liste ordonnée de sous arbres imbriqués, allant de l'arbre maximum jusqu'au sous arbreG_R qui ne contient que la racine :

$$G_{max} < G_1 < G_2 < .... < G_t < ... < G_R$$

Cet ordre résulte du critère « coût de complexité ».

La seconde étape consiste à déterminer, dans cette liste le sous arbre optimal au sens d'un second critère qui est le taux d'erreur en validation. Le sou arbre G fournit le taux minimum d'erreur en validation. L'élagage s'effectue par le bas en remontant jusqu'à la racine.

6.1. Construction de la liste des sous arbres

Elle s'effectue en deux phases :

La première permet d'élaguer les sous arbres qui ont deux sommet terminaux et dont la suppression ne provoque aucune augmentation du critère de segmentation. La seconde est répétitive, elle consiste à évaluer « le coût de complexité » de chacun des sous arbres restant afin de déterminer celui qu'il supprimer. Nous allons décrire de manière plus explicite de ces deux phases :

Phase initiale : soit S_{ter}, l'ensemble des sommets terminaux de la partition, et soient t_{jg} ett_{id} deux d'entre eux, ils ont le même sommet père t ;. On sait que la valeur du critère de segmentation est toujours positive ou nulle $\triangle_i(d,t) \geqslant 0$.

Si elle est nulle, cela signifie que la segmentation de t, en deux segments n'apporte aucune information sur les classes à discriminer.

Plus généralement, considérons que nous somme à l'itération V, dans laquelle la partition terminale est $Ster^v$ est soit t_u un sommet intermédiaire dont les fils sont terminaux :$t_{u,g}, t_{u.d}$. Si $\triangle i(d,t) \doteq 0$alors on peut supprimer de la partition terminale les deux sommets ainsi tu deviendras un sommet terminal.

Phase courante : à l'issue de la phase initiale nous obtenons une partition terminale que nous noterons S^1_{ter} le tout, formant un premier sous arbre binaire noté G_o, il s'agit maintenant de passer à un nouveau sous arbre binaire $G1$. Pour le déterminer nous utiliserons une procédure différente.

Le coût de complexité d'un arbre binaire est une combinaison linéaire entre le taux d'erreur de la partition et le nombre de sommets qui la composent. L'objectif est de déterminer un sous arbre G_i qui a le plus faible « coût complexité » :

$C\alpha(G) \doteq \xi(S_{ter}) + (S_{ter}) pour \alpha \epsilon [0,1]$

Pour chaque valeur de α un sous arbre qui a le plus faible « coût de complexité » sera déterminer.

Plaçons nous dans le cas du passage de G_o à G_i. Considérons un sommet$t \epsilon S^1_{ter}$ non terminal de G_o nous calculerons alors deux quantités :

1. le coût de complexité en t est $C\alpha(t) \doteq \zeta(t) + \alpha$ qui n'est rien d'autre que le taux d'erreur au sommet t :

$$\xi(t) \doteq 1 - max$$

$$\xi(G_k) = \min_{G \in S_B} \xi(G)$$

$$\xi(t) = \max_{r=1}^{k} p(\frac{r}{t})$$

2.le coût de complexité du sous arbre $G(t^+)$ qui a pour racine t

$$C_\alpha(G) = \xi(S_t^+) + (S_t^+)$$

Ou S_t^+ est la partition terminal associée au sous arbre $G(t^+)$
Il est évidant que si :

$$C_\alpha(t) < C_(t^+)$$

alors ,il vaut mieux d'élaguer le sous arbre $G(t^+)$,mais tant que :

$$C_\alpha(t) > C_(t^+)$$

Il est préférable de garder $G(t^+)$ En faisant varier ce par valeur croissante, il est clair que l'inégalité $C_\alpha(t) > C_(t^+)$ va à une valeur précise de α , elle sera :$C_\alpha(t) > C_(t^+)$.

Ainsi nous obtenons :

$$\alpha = \frac{\zeta(t) - \zeta(S_t^+}{card(S_t^+) - 1}$$

Pour tout t e s . nous déterminons ait). m Puisque nous cherchons la seconde valeur de a qui nous permet d'avoir le sous arbre Gi, il faut alors prendre :

$$\alpha_1 = \min_{t \epsilon S_{int}^1)}$$

Soit $t \epsilon S_{int}^1$ que $\alpha(t) = \alpha_1$, le sous arbre G_1est obtenu en retirerons à G_o le sous arbre $G(t^+)$. Dans le cas où il y a plusieurs sommet $t_k k = l, ..., k$ qui vérifient$\alpha(t_k) = \alpha_1$, nous retirons tous les sous arbres $G(t^+)$ dont ils sont racine. Nous recommençons le processus en cherchant cette fois-ci, G$_2$à$partir de G_1 et ainsi de suite, jusqu'$à$la racine$.

Nous obtenons ainsi une liste de couples (a ?Gj) que nous ordonnons suivant les valeurs de α_i :

$$G_o < G_1 < ... < G_t < ... < G_R$$

6.2. Détermination du meilleur sous arbre G

Nous choisirons parmi les sous arbres de la liste ordonnée le meilleur selon un critère. La procédure la plus simple consiste à disposer d'un échantillon test Ω, et évaluer le taux d'erreur lors du classement des individus de cet échantillon à partir de chaque sous arbre. Soit$\zeta_t(G_i)$, le taux d'erreur sur l'échantillon de test Ωlorsqu'on applique le modèle de décision donnée par G_i,le meilleur sous arbreG^* est tel que :

$$\xi(G_i) = \min_{i=0,R} \xi(G_i)$$

6.3. Spécialisation de l'arbre

La spécialisation de l'arbre consiste a étiqueter chacun de ses sommets par l'une des classes $C_1, C_2, ..., c_m$. La stratégie la plus couramment utilisée revient à affecter un sommet à la classe majoritaire. Si on désigne par $n_i S/n_s$ la proportion d'individus de l'échantillon d'apprentissage qui appartiennent à la classeC_i du sommets s , celui-ci sera alors associé à la classeC_k, si :

$$\frac{n_{is}}{n_s} = \max_{i=0}^{m}(\frac{n_{is}}{n_s})$$

cette spécialisation permet d'affecter un nouvel individu, n'ayant pas servi l'ors de la phase d'apprentissage, à un des classes $C_1, C_2, ..., C_m$

CHAPITRE IV

RÉSEAUX NEURONES

Les réseaux de neurones se sont imposés dans plusieurs domaines durant ces dernières années comme un outil universel. En statistique ils sont utilisés en tant que classificateurs (analyse discriminante), détecteurs de classes (classification automatique), estimateurs non-paramétrique de régression non linéaire et comme estimateurs de fonctions de densité Les réseaux de neurones sont composés d ?éléments simples (ou neurones). Ces éléments ont été fortement inspirés par le système nerveux biologique.Comme dans la nature, le fonctionnement du réseau (de neurone) est fortement influencé par la connections des éléments entre eux. On peut entraîner un réseau de neurone pour une tâche spécifique (reconnaissance de caractères par exemple) en ajustant les valeurs des connections (ou poids) entre les éléments (neurones). En général, l ?apprentissage des réseaux de neurones est effectué de sorte que pour une entrée particulière présentée au réseau corresponde une cible spécifique. L'ajustement des poids se fait par comparaison entre la réponse du réseau (ou sortie) et la cible, jusqu'à ce que la sortie corresponde (au mieux) à la cible. On utilise pour ce type d'apprentissage dit superviser un nombre conséquent de pair entrée/sortie.

L'apprentissage « par paquet » (batch training) du réseau consiste à ajuster les poids et biais en présentant les vecteurs d'entrée/sortie de tout le jeu de données.

L'apprentissage « pas à pas ou séquentiel » (incremental training) consiste à ajuster les poids et biais en présentant les composantes du vecteur d'entrée/sortie les unes après les autres. Ce type d'apprentissage est souvent qualifié d'apprentissage « en ligne » (« on line » training) ou « adaptatif » (« adaptive » training). L'apprentissage permet aux réseaux de neurones de réaliser des taches complexes dans différents types d'application (classification, identification, reconnaissance de caractères, de la voix, vision, système de contrôle ?). Ces réseaux de neurones peuvent souvent apporter une solution simple à des problèmes encore trop complexes ne pouvant être résolus rapidement par les ordinateurs actuels (puissance de calcul insuffisante) ou par notre manque de connaissances. La méthode d'apprentissage dite supervisé est souvent utilisée mais des techniques d'apprentissage non supervisé existent pour des réseaux de neurones spécifiques. Ces réseaux peuvent, par exemple, identifier des groupes de données (réseaux de Hopfield). Les réseaux de neurones ont une histoire

relativement jeune (environ 50 ans) et les applications intéressantes des réseaux de neurones n'ont vu le jour qu'il à une vingtaine d ?année (développement de l'informatique).

I. Le Neurone Formel

Il a été présenté par McCulloch et Pitts en 1943 [30-26][24], c'est une cellule ayant plusieurs entrées et une sortie. Pour chaque entrée, un poids synaptique lui est associé représentant ainsi la force de connexion entre cette entrée et le neurone. Ces entrées proviennent des neurones en amont et sa sortie alimente les neurones en aval [18]. La figure illustre un neurone formel.

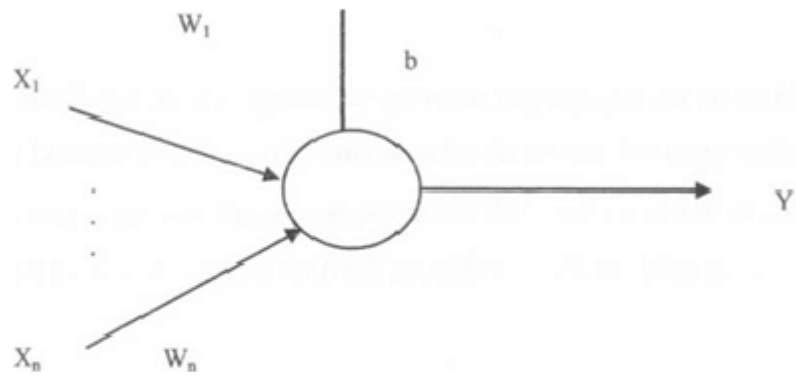

FIGURE IV.1 – Le neurone formel

Avec :
X : la iéme entrée.
n : le nombre d'entrées.
W ! : le poids synaptique de la ieme entrée.
b : la fonction seuil.
Y : la sortie du neurone.

Le neurone formel, est considéré comme une modélisation élémentaire du neurone réel, c'est un automate possédant n entrées réelles $X_1, ..., X_n$, et dont le traitement

consiste à affecter à sa sortie Y, le résultat d'une fonction d'activation/de la somme pondérée de ses entrées.

II. le perceptron :

1. Quelques définitions :

-a- Le neurone

Un neurone est l'unité élémentaire de traitement d'un réseau de neurones. Il est connecté à des sources d'information en entrée (d'autres neurones par exemple) et renvoie une information en sortie. Voyons comment tout cela s'organise.

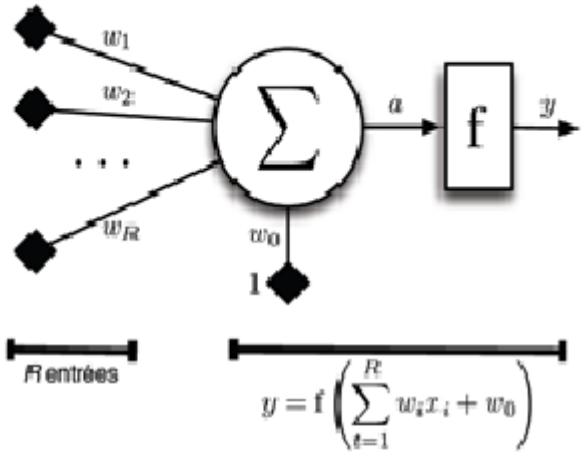

FIGURE IV.2 – Un neurone

-b- Les entrées :

On note $(X_i)_1 \leq i \leq n)$ les n informations parvenant au neurone. De plus, chacune sera plus ou moins valorisée vis à vis du neurone par le biais d'un poids. Un poids est simplement un coefficient $W_i$lié à l'information X_i. La i-ème information qui parviendra au neurone sera donc en fait $W_i * X_i$. Il y a toutefois un "poids" supplémentaire, qui va représenter ce que l'on appelle le coefficient de biais. Nous le noterons W_0 et le supposerons lié à une information $X_0 = -1$. Nous verrons plus

tard son utilité, dans la section fonction d'activation . Le neurone artificiel (qui est une modélisation des neurones du cerveau) va effectuer une somme pondérée de ses entrées plutôt que de considérer séparément chacune des informations. On définit une nouvelle donnée in, par :

$$in = \sum_{i=0}^{n} w_i \times x_i = \Big(\sum_{i=0}^{n} w_i \times x_i \Big) - w_0$$

C'est en fait cette donnée-là que va traiter le neurone. Cette donnée est passée à la fonction d'activation, qui fait l'objet de la prochaine section. C'est d'ailleurs pour ça que l'on peut parfois appeler un neurone une unité de traitement[32]

-c- La fonction d'activation :

La fonction d'activation, ou fonction de transfert, est une fonction qui doit renvoyer un réel proche de 1 quand les "bonnes" informations d'entrée sont données et un réel proche de 0 quand elles sont "mauvaises". On utilise généralement des fonctions à valeurs dans l'intervalle réel $[0, 1]$. Quand le réel est proche de 1, on dit que l'unité (le neurone) est active alors que quand le réel est proche de 0, on dit que l'unité est inactive. Le réel en question est appelé la sortie du neurone et sera noté a. Si la fonction d'activation est linéaire, le réseau de neurones se réduirait à une simple fonction linéaire. En effet, si les fonctions d'activations sont linéaires, alors le réseau est l'équivalent d'une régression multi-linéaire (méthode utilisée en statistiques). L'utilisation du réseau de neurone est toutefois bien plus intéressante lorsque l'on utilise des fonctions d'activations non linéaires. En notant f la fonction d'activation, on obtient donc la formule donnant la sortie d'un neurone :

$$a = f(in) = f\Big(\sum_{i=0}^{n} w_i \times x_i \Big)$$

Remarquons que le coefficient de biais est inclus dans la somme, d'où la formule plus explicite :

$$a = f(in) = f\left(\Big(\sum_{i=0}^{n} w_i \times x_i \Big) - w_0 \right)$$

Il y a bien sûr beaucoup de fonctions d'activations possibles, c'est à dire répondant aux critères que nous avons donnés, toutefois dans la pratique il y en a principalement trois qui sont utilisées :

La fonction à seuil, définie par :$f(x) = 1$ si $x \geqslant 0$et0 sinon

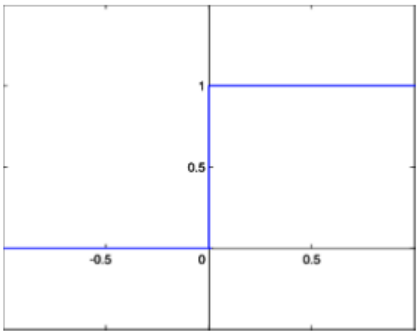

FIGURE IV.3 – Graphe de la fonction à seuil

La fonction logistique, définie par :$f(x) = \dfrac{1}{1+e^{-x}}$

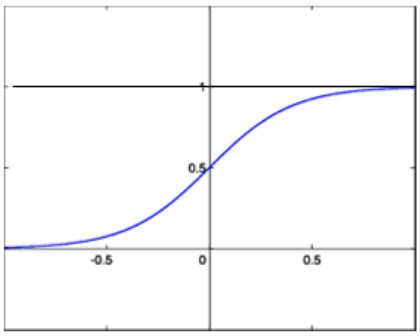

FIGURE IV.4 – Graphe de la fonction logistique

La fonction tangente hyperbolique, définie par : $f(x) = \dfrac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$

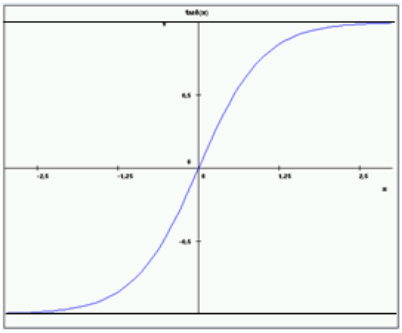

FIGURE IV.5 – Graphe de la fonction tangente hyperbolique

Remarques :

- la fonction à seuil est une Fonction non dérivable et non continue.
- Les fonctions logistique et tangente hyperbolique présentent l'avantage d'être dérivable (ce qui va être utile par la suite) ainsi que de donner des valeurs intermédiaires (des réels compris entre 0 et 1) par opposition à la fonction de à seuil qui elle renvoie soit 0 soit 1.
- Toutefois, les deux fonctions possèdent un seuil. Celui de la fonction à seuil est en $x = 0$ et vaut 1 alors que celui de la fonction logistique est en 0 également mais vaut 1/2.
  - La fonction logistique :

$$f(x) = \frac{1}{1 + e^{-x}} \quad , \quad f'(x) = f(x)\Big(1 - f(x)\Big)$$

  - La fonction tangente hyperbolique :

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad , \quad f'(x) = f(x)\Big(1 + f(x)\Big)\Big(1 - f(x)\Big)$$

  Revenons à notre neurone et demandons-nous quand est-ce que le seuil est atteint, ou dépassé dans le cas de la fonction sigmoïde. Il est dans tous les cas atteint quand in vaut 0.

$$\begin{aligned} in = 0 \quad &\iff \quad \sum_{i=0}^{n} w_i \times x_i = 0 \\ &\iff \quad \Big(\sum_{i=0}^{n} w_i \times x_i\Big) - w_0 = 0 \\ &\iff \quad \sum_{i=0}^{n} w_i \times x_i = w_0 \end{aligned}$$

C'est là qu'intervient réellement le coefficient de biais. Nous voyons donc que l'on atteint le seuil de la fonction d'activation lorsque la somme pondérée des informations d'entrée vaut le coefficient de biais . De plus :

$$in \geqslant 0(in) \geqslant seuil$$

Maintenant, notons $W = (W_i)_{1 \leq i \leq n}$ le vecteur dont les composantes sont les poids et $X = (X_i)_{1 \leq i \leq n}$ le vecteur dont les composantes sont les informations d'entrées du neurone.

Avec cette notation on obtient :$W \times X = 0$ Ceci définit un hyperplan d'un espace de dimension n . En fait, l'espace dont il est question est l'espace des informations d'entrées. De la même manière que dans notre monde nous décrivons des points avec nos 3 coordonnées souvent notées (x, y, z), dans l'espace des informations d'entrée on note les coordonnées$(\mathrm{X}_1, ..., X_n)$.

Un hyperplan est un espace de dimensionn-1 . Dans notre monde, l'espace est de dimension 3 (car 3 coordonnées) et un hyperplan est donc un espace de dimension 2. En dimension 3, un hyperplan est donc simplement un plan. En dimension 2, un hyperplan est par conséquent une droite. Plaçons-nous en dimension 2. Tracez donc 2 axes perpendiculaires. Maintenant, tracez une droite. Vous voyez que cette droite sépare le plan en 2 parties. A quoi cela sert-il ? En fait, un réseau de neurone simple va permettre de classer les points du plan dans une partie ou l'autre du plan grâce à cette droite, dans le cas de la dimension 2. Encore une fois, dans ce cas-là, ceux qui seront dans une partie du plan appartiendront à la première classe et ceux qui seront dans l'autre partie du plan (de l'autre côté de la droite) appartiendront à la deuxième classe.

-e- Activation et condition d'activation :
On dit que le neurone est actif lorsque$(in \geqslant 0)$, autrement dit lorsque$\alpha = f(in)$. Similairement, on dit que le neurone est inactif lorsque$(\mathrm{in} \leqslant 0), autrementditlorsque\alpha = f(in) \leqslant seuil$.

Définition :
Le perceptron peut être vu comme le type de réseau de neurones le plus simple. C'est un classifieur linéaire. Ce type de réseau neuronal ne contient aucun cycle (en

anglais feedforward neural network). Dans sa version simplifiée, le perceptron est mono-couche et n'a qu'une seule sortie à laquelle toutes les entrées sont connectées.

> Un réseau de neurones monocouche, aussi appelé perceptron, est caractérisé de la manière suivante :Il possède n informations en entrée ; Il est composé de p neurones, que l'on représente généralement alignés verticalement. Chacun peut en théorie avoir une fonction d'activation différente. En pratique, ce n'est généralement pas le cas. Chacun des$p$neurones est connecté aux n informations d'entrée.

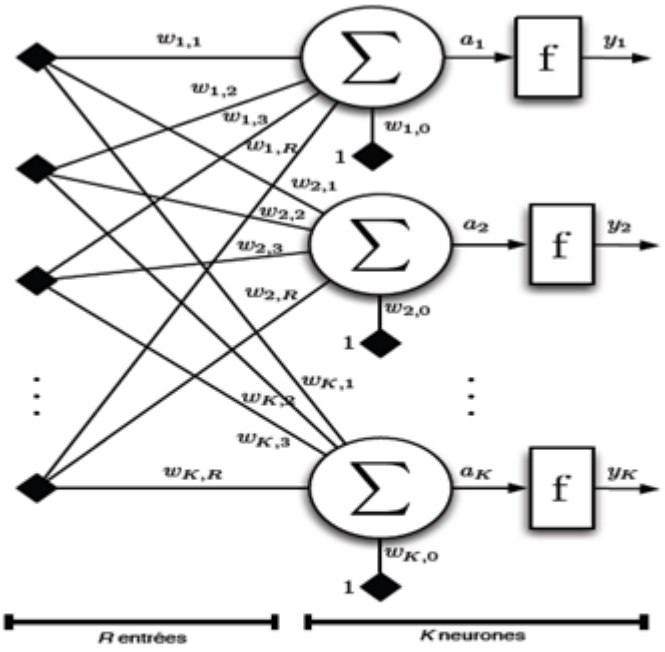

FIGURE IV.6 – Perceptron

Le réseau de neurones possède ainsi n informations en entrée et$p$ sorties, chaque neurone renvoyant sa sortie.

Pour la suite, on notera :

- $X = (x_i), 1 \leqslant i \leqslant n$: les$n$ informations d'entrée ;
- $W_{i,j}$ et $1 \leqslant i \leqslant n$ et $1 \leqslant j \leqslant p$: le poids reliant l'informationx_i et le neurone j puis a_jl'activation du j -ème neurone ;
- $W_{0,j}$: le coefficient de biais, également appelé seuil, du j -ème neurone ;
- in_j : la donnée d'entrée (somme pondérée) du j -ème neurone.

On a donc l'équation suivante :

$$\forall 1 \leq j \leq p \ \ , \ \ a_j = f(in_j) = f\Big(\sum_{i=1}^{n} w_{i,j} \times x_i\Big) = f\bigg(\Big(\sum_{i=1}^{n} w_{i,j} \times x_i\Big) - w_{0,j}\bigg)$$

Chaque neurone de la couche donnera donc une sortie. Une utilisation courante est que chaque neurone de la couche représente une classe. Pour un exemple$X$ donné, on obtient la classe de cet exemple en prenant la plus grande des p sorties. Essayons maintenant d'approfondir ce modèle.

2. Modélisation Matricielle :

Avec la notation que nous avons définie pour les poids, on obtient ainsi une matrice $W = (W_{i,j})1 \leqslant i \leqslant net1 \leqslant j \leqslant p$ où $W_{i,})$ représente le coefficient liant la i -ème information au j -ème neurone. si on représente par$A = (a_j), 1 \leqslant j \leqslant p$ la liste des p sorties du réseau de neurone, on obtient l'équation matricielle suivante, en adaptant la fonction de transfert à notre nouvelle modélisation : on définit une nouvelle fonction f qui prend l'image d'un vecteur en calculant l'image de chacune de ses composantes, et retourne un vecteur dont les composantes sont les images des composantes de départ.

$$A = f(W^t \times X)$$

Cette modélisation n ?est pas fréquemment utilisée mais représenter un réseau de neurones monocouche par une matrice permet d ?avoir une autre vision du problème.

3. Les différents types de perceptrons :

Il existe 2 types de perceptrons : les perceptrons feed-forward et les perceptrons récurrents . Les perceptrons récurrents sont ceux qui alimentent leurs entrées avec leurs sorties, alors que les perceptrons feed-forward non.

4. Comment choisir les poids ?

[28] Il existe plusieurs algorithme(s) qui permettent à un perceptron d'adapter ses poids à un ensemble d'exemples de sorte à obtenir pour cet ensemble la classification attendue. Ainsi, si l'ensemble d'exemples est assez vaste (les exemples sont assez variés), on pourra obtenir un perceptron qui donnera des résultats convenables pour des exemples non rencontrés.

5. Apprentissage du perceptron :

Il y a deux algorithmes, principalement, pour "faire apprendre" à un réseau de neurones monocouche. Le premier est la méthode simple et se nomme la

descente de gradient . L'autre, un peu plus efficace généralement, se nomme algorithme de Widrow-Hoff , du nom des deux scientifiques qui ont élaboré cette technique. Les deux méthodes consistent à comparer le résultat qui était attendu pour les exemples puis à minimiser l'erreur commise sur les exemples. Toutefois, il existe bien sûr une nuance entre les deux méthodes, qui va être expliquée plus loin. Nous allons, pour chacune des méthodes, étudier la correction des poids concernant seulement l'un des neurones. Il suffira d'appliquer successivement la méthode à chacun des neurones du réseau monocouche.

5.1. Apprentissage par descente de gradient :

Pour comprendre cette méthode d'apprentissage, il faut définir l'erreur quadratique E. Si l'on est en présence de N exemples, alors pour 1, notons (X_k, Y_k) le couple exemple - sortie attendue , où $X_k = (x_i), 1$ est le vecteur dont les coordonnées sont les$n$ informations d'entrée de l'exemple et Y_k est la sortie attendue (la sortie "vraie") pour cet exemple-là de la part de notre neurone. Enfin, on note S_k la sortie obtenue pour le k- ème exemple avec les poids actuels. Alors, l'erreur quadratique est définie comme suit.

$$E = \frac{1}{2}\sum_{k=1}^{n}\Big(Y_k - S_k\Big)^2$$

On voit donc que l'erreur est nulle si le réseau de neurones ne se trompe sur aucun des exemples, c'est à dire s'il parvient à calculer la bonne sortie (par exemple classifier) pour chacun des exemples correctement. C'est rarement le cas car souvent on démarre avec des poids tirés aléatoirement. Il s'agit donc de minimiser, pour un ensemble de N exemples donné, cette erreur quadratique. On va noter ? un nombre réel auquel on donne le nom de taux d'apprentissage. C'est nous qui devons lui donner une valeur lors de la mise en pratique de l'apprentissage. Comme nous ne considérons qu'un neurone à la fois, on va noter W_i le poids reliant la i -ème information à notre neurone. La méthode de descente du gradient consiste en fait à effectuer les actions suivantes :

Etapes de la méthode de descente du gradient :

- Créer n variables $dW?_i$ pour$1 \leqslant i \leqslant n, égales à 0$
- Prendre un exemple e_k , pour$1 \leqslant k \leqslant N$
- Calculer la sortie obtenue avec les poids actuels, notée $?s_k$ (1)
- Rajouter à dW_i , pour tout $1 \leqslant i \leqslant n$, le nombre $(y_k - s_k)x_i$ (2)
- Réitérer (1) et (2) sur chacun des exemples
- Pour $1 \leqslant i \leqslant n$, remplacer W_i par $W_i + dW_i$

Afin d'obtenir de bons résultats, il faudra passer plusieurs fois les exemples à chaque neurone, de sorte que les poids convergent vers des poids "idéaux". Le problème avec cette méthode est que l'on corrige sur la globalité des exemples, ce qui fait que le réseau ne s'adaptera aux exemples qu'après un certain moment. Il y a une autre méthode qui permet de corriger sur chacun des exemples, et qui se nomme méthode d'apprentissage de Widrow-Hoff.

5.2. Apprentissage par l'algorithme de Widrow-Hoff :

L'algorithme de Widrow-Hoff, ou encore "la règle delta", n'est en fait qu'une variante de l'algorithme précédent.

Comme nous l'avons vu dans l'algorithme précédent, on engrange les erreurs commises sur chaque exemple puis pour terminer on corrige le poids, et ce pour chaque poids. Vous ressentez probablement cette méthode comme assez "grossière", dans le sens où elle n'est pas très précise et met beaucoup de temps avant de tendre vers le bon coefficient. On ressent le fait qu'elle ne corrige qu'un petit peu alors qu'elle pourrait corriger beaucoup mieux pour chaque exemple. Et c'est là que l'algorithme de Widrow-Hoff intervient. En effet, la méthode élaborée par Widrow et Hoff consiste à modifier les poids après chaque exemple, et non pas après que tous les exemples aient défilé. Ceci va donc minimiser l'erreur de manière précise, et ce sur chaque exemple. Instinctivement, on constate bien que le réseau de neurones va s'améliorer nettement mieux et va tendre bien plus rapidement à classifier parfaitement (ou presque) chacun des exemples, bien que des méthodes plus efficaces encore existent.

Cette méthode est bien sûr plus efficace, comme dit précédemment, mais l'algorithme est plus simple également.

Appliquer plusieurs fois cet algorithme permettra d'affiner la correction d'erreur et d'obtenir un réseau de neurones de plus en plus performant. Attention toutefois, car l'appliquer un trop grand nombre de fois mènerait à ce que l'on appelle "l'overfitting" (sur-apprentissage), c'est à dire que votre réseau devient très performant sur les exemples utilisés pour l'apprentissage, mais ne parvient peu ou pas à généraliser pour des informations quelconques.

III. Perceptron multicouche :

1. Définition :

Nous avons précédemment étudié les perceptrons (réseaux monocouche) et nous avons vu que les neurones de sortie étaient chacun connectés aux mêmes informations. Nous les avons perçus comme une couche (alignés verticalement). Ainsi, une couche est constituée de neurones étant connectés aux mêmes informations mais n'étant pas connectés entre eux. Il s'agit maintenant de généraliser le perceptron. On peut ainsi disposer les neurones en plusieurs couches. Ainsi les informations en entrée sont connectés à tous les neurones de la première couche, tous les neurones de la première couche sont connectés à tous les neurones de la seconde couche, et ainsi de suite jusqu'à la dernière couche, appelée couche de sortie. Toutes les couches exceptée la

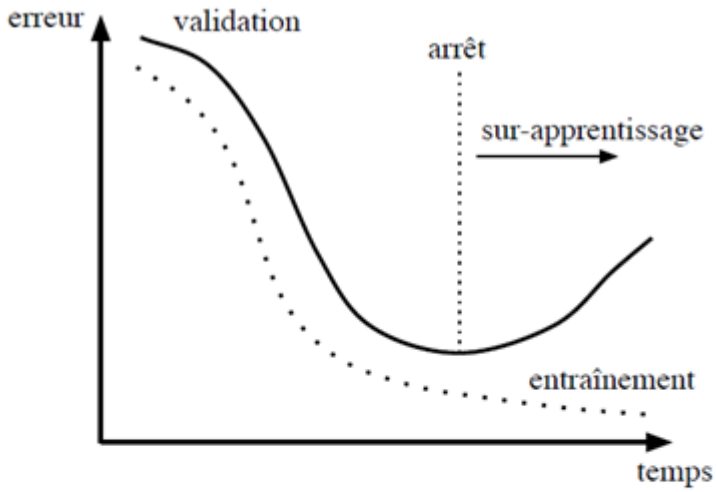

FIGURE IV.7 – Illustration de sur-apprentissage

couche de sortie sont considérées comme "couches cachées ".

2. Apprentissage du perceptron multicouches

: De la même manière que le perceptron monocouche, le perceptron multicouche est lui aussi capable d'apprentissage. En effet, il existe également un algorithme permettant de corriger les poids vis à vis d'un ensemble d'exemples donnés

Algorithme d'apprentissage par descente du gradient :
Le principe de l'algorithme est de minimiser une fonction d'erreur. Il s'agit ensuite de calculer la contribution à cette erreur de chacun des poids synaptiques. En effet, chacun des poids influe sur le neurone correspondant, mais la modification pour ce neurone va influer sur tous les neurones des couches suivantes.

Soit un perceptron multicouches défini par une architecture à $n$entrées et à p sorties, soit $\vec{w}$ le vecteur des poids synaptiques associés à tout les liens du réseau.

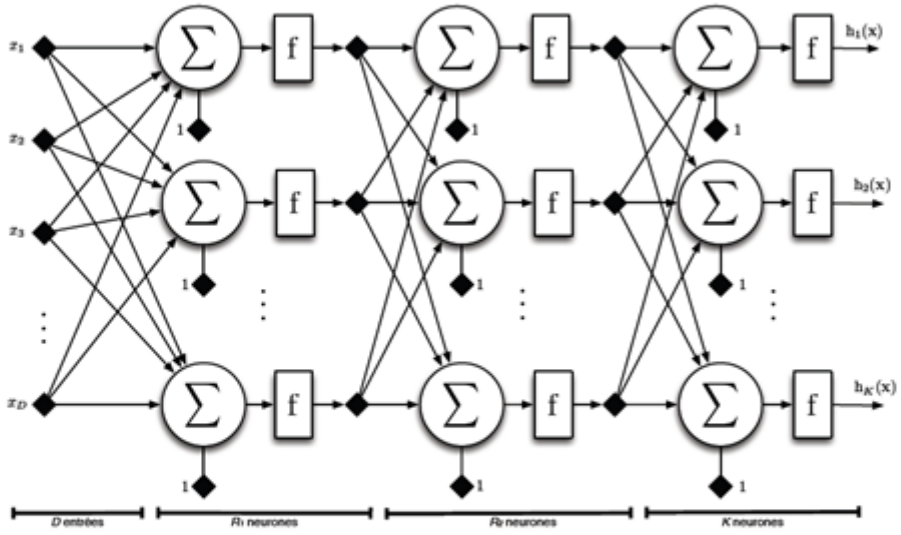

FIGURE IV.8 – Perceptron multicouche

L'erreur du Perceptron multicouche sur un échantillon d'apprentissage$S$ d'exemples(O^s, Z^s) est définie par :

$$E(\vec{w}) = \frac{1}{2} \sum_{(O^s,Z^s)\in S} \sum_{k=1}^{p} \left(O_k^s - y_k^s\right)^2$$

L'erreur mesure donc l'écart entre les sorties attendues et calculées sur l'échantillon complet. On suppose S fixé, le problème est donc de déterminer un vecteur $\vec{w}$ qui minimise$E(\vec{w})$. Cependant, on cherche à minimiser ' ?erreur sur chaque présentation individuelle d'exemple. L'erreur pour un exemple est :

$$E_{(O^s,z^s)}(\vec{w}) = \frac{1}{2} \sum_{k=1}^{p} \left(O_k^s - y_k^s\right)^2 \qquad \text{(IV. 1)}$$

La méthode du gradient peut être étendue au cas de fonctions de plusieurs variables réelles. Pour mettre enœ uvre la méthode appliquée, nous allons tout d'abord, évaluer la dérivée partielle de E par rapport à $\vec{w}$, pour tout i. On a :

$$\frac{\partial E}{\partial W_{i,j}} = \frac{\partial E}{\partial S_i}\ \frac{\partial S_i}{\partial W_{i,j}} = \frac{\partial E}{\partial S}\ Z_{i,j}$$

Il suffit de calculer $\frac{\partial E}{\partial S_i}$, pour cela nous allons distinguer deux cas :
– Le cas où la couche i est une couche de sortie.
– Le cas où la couche i est une couche interne.
Si i est une couche de sortie, la quantitéS_i ne peut influencer la sortie du réseau que par le calcul de y_i.

On a donc

$$\frac{\partial E}{\partial S_i} = \frac{\partial E}{\partial y_i}\ \frac{\partial y_i}{\partial E}$$

Or,

$$\frac{\partial E}{\partial y_i} = \frac{\partial}{\partial y_i}\ \frac{1}{2}\sum_{k=1}^{p}\Big(O_k - y_k\Big)^2$$

Seul le terme correspondant à $k = i$ a une dérivée non nulle, ce qui nous donne finalement :

$$\frac{\partial E}{\partial y_i} = \frac{\partial}{\partial y_i}\ \frac{1}{2}\Big(O_k - y_k\Big)^2 = -(O_i - y_i)$$

Pour le second terme de la dérivée de l'équation IV. 1 , en utilisant la formule de calcul de la dérivée de la fonction sigmoïde, nous avons :

$$\frac{\partial y_i}{\partial S_i} = y_i(1 - y_i)$$

D'où

$$\frac{\partial E}{\partial S_i} = -(O_i - y_i)y_i(1 - y_i)$$

Si i est une couche cachée, dans ce cas S_i va influencer le réseau par tout les calcules des neurones de la couche de sortie. Nous avons alors :

$$\frac{\partial E}{\partial S_i} = \sum_k \frac{\partial E}{\partial S_k}\ \frac{\partial S_k}{\partial S_i} = \sum_k \frac{\partial E}{\partial S_k}\ \frac{\partial S_k}{\partial y_i}\ \frac{\partial y_i}{\partial S_i}$$

Et

$$\frac{\partial E}{\partial S_i} = \sum_k \frac{\partial E}{\partial S_k}\ \times\ w_{i,j}\ \times y_i(1 - y_i)$$

D'où

$$\frac{\partial E}{\partial S_i} = y_i(1 - y_i)\sum_k \frac{\partial E}{\partial S_k}\ \times\ w_{i,j}$$

Enfin pour en déduire la modification à effectuer sur les poids synaptiques, il nous reste simplement à rappeler que la méthode du gradient nous indique que :

$$\Delta w_{i,j} = -\varepsilon \frac{\partial E(\vec{w})}{\partial w_{i,j}}$$

Avec ε un paramètre de proportionnalité.

Conclusion :

Les réseaux de neurones sont des outils statistiques, qui permettent d'ajuster des fonctions Non linéaires très générales à des ensembles de points ; comme toute méthode statistique, L ?utilisation de réseaux de neurones nécessite que l'on dispose de données suffisamment Nombreuses et représentatives.
Il est toujours souhaitable, et souvent possible, d'utiliser, pour la conception du réseau, les Connaissances mathématiques dont on dispose sur le phénomène à modéliser.
Il y a principalement deux facteurs qui influent sur l'apprentissage. Ce sont la qualité de l'échantillonnage d'apprentissage(les exemples qui constituent la base d'apprentissage) et la diversité des valeurs. En effet, le réseau de neurones généralisera mieux (aura plus de chances de répondre correctement en lui donnant en entrée des informations non présentes dans les exemples d'apprentissage) si la qualité de l'échantillonnage est meilleure et si les données des exemples d'apprentissage sont variées. Intuitivement, on est conscient que s'il sait répondre Correctement pour un nombre fini de situations les plus diverses, il sera alors plus proche de ce que l'on veut dans une situation nouvelle.
Il existe plusieurs algorithme(s) qui permettent à un perceptron d'adapter ses poids à un ensemble d'exemples de sorte à obtenir pour cet ensemble la classification attendue. Ainsi, si l'ensemble d'exemples est assez vaste (les exemples sont assez variés), on pourra obtenir un perceptron qui donnera des résultats convenables pour des exemples non rencontrés.

CHAPITRE V

APPLICATION

Analyse en composantes principales

Nous allons nous intéresser dans ce chapitre à l'analyse en composantes principales des données dont nous disposons. L' ACP est une méthode statistique multidimensionnelle qui permet de synthétiser un ensemble de données en identifiant la redondance dans celles-ci et consiste à rechercher les directions de l'espace qui représentent le mieux les corrélations entre p variables. C'est une méthode très utilisé comme méthode descriptive, permet de visualiser l'information contenue dans les tableaux de données quantitatives. Cette méthode permet de traiter simultanément un nombre quelconque de variables, toutes quantitatives. Elle consiste en la diagonalisation de la matrice de variances-covariances. La première valeur propre (associée à ce premier axe) mesure la quantité d'information présente le long de cet axe. On analyse ainsi les différents axes, en reconstituant progressivement la totalité des données.
L'analyse en composantes principales est une technique statistique permet, quand on dispose d'une population d'individus pour lesquelles on possède de nombreux renseignements concernant les pratiques alimentaires et le statut, d'en donner une représentation géométrique, c'est-à-dire en utilisant un graphique qui permet de voir les rapprochements et les oppositions entre les caractéristiques des individus.

Nous partons d'un tableau de données croisant 4812 individus et 21 variables qui sont l'âge et la consommation de « painjc, couscouljc, patejc, rizjc, legumejc, fruitjc, pomterrjc, legusecjc, laitjc, poissonjc, viandejc, volailljc,oeufjc, cacahuejc, dessertjc, huilolijc, huilejc, beurrejc, feculenj,cerealej».

3. matrice de corrélation

D'après la matrice de corrélation présentée en annexe :

Nous remarquons que
- les fortes corrélations sont toutes positives.
- La variable feculenj est corrélée avec presque toutes les variables et fortement corrélée avec painjc.

Mise à part la variable feculenj.

- Les autres corrélations sont peu élévées. Les plus fortes sont entre fruitj et legumejc ; patejc et rizjc ; pomterr et legumejc.
- On remarque également que la variable "age" est faiblement corélée avec toutes les autres variables.

	F1	F2	F3	F4	F5	F6
Valeur propre	3,603	1,890	1,568	1,156	1,102	1,066
Variabilité (%)	17,157	9,000	7,465	5,504	5,246	5,078
% cumulé	17,157	26,156	33,621	39,126	44,371	49,450

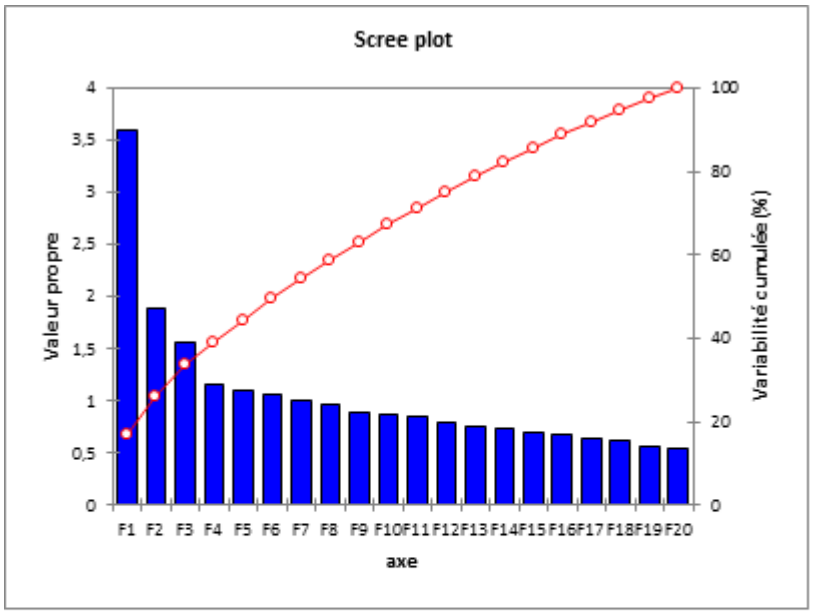

FIGURE V.1 – les valeurs propres

Les vecteurs propres

Tableau *partiel* des vecteurs propres, le tableau complet se trouve en annexe page[116]

	F1	F2	F3
painjc	**0,264**	**-0,291**	**0,355**
couscou1jc	**0,214**	**0,340**	0,171
patejc	**0,224**	**0,364**	0,147
rizjc	**0,255**	0,276	0,088
cerealjc	0,185	0,345	-0,112
pomterrjc	**0,259**	-0,238	**0,234**
legusecjc	0,177	**0,299**	-0,071
fruitjc	**0,221**	-0,152	**-0,386**
legumejc	**0,264**	**-0,203**	-0,126
laitjc	0,175	**-0,296**	-0,090
poissonjc	0,108	0,004	-0,200
viandejc	0,200	0,156	**-0,335**
volailljc	0,198	0,019	**-0,268**
oeufjc	0,180	-0,126	**-0,249**
cacahuej	0,108	0,140	-0,164
dessertjc	**0,220**	-0,191	-0,219
huilolijc	0,003	-0,200	-0,137
huilejc	**0,232**	-0,166	0,121
beurrejc	**0,246**	0,070	-0,092
age	-0.032	0,033	0,102
feculenj	**0,435**	-0,062	0,396

L'interprétation des axes

Le premier axe factoriel

Le premier axe factoriel est expliqué par les variables suivantes : painjc, couscou1jc, patejc, rizjc, pomterrjc, fruitjc, legumejc, dessertjc, huilejc, beurrejc et surtout feculenj
Le signe étant positif pour toutes ces variables, c'est un axe d'échelle. Il ordonne en projection, les individus de droite à gauche en commençant par les individus (individus situés à droite de 0) qui consomment en grandes valeurs de {painjc, couscou1jc, patejc, rizjc, pomterrjc, fruitjc, legumejc, dessertjc, huilejc, beurrejc, feculenj} en finissant par la projection des individus ayant les plus faibles valeurs de consommation de {ainjc, couscou1jc, patejc, rizjc, pomterrjc, fruitjc, legumejc, dessertjc, huilejc, beurrejc, feculenj} (individus situés à gauche de 0).
Le nombre d'individus étant de 4812, il est difficile de détecter quels sont les individus qui se projettent à droite et quels sont ceux qui se projettent à gauche. Néanmoins on peut remarquer certains individus comme {330, 4395, 4651, 385, 347, 355, 4643, 352, 4514, 3912, 352, 328, 3900, 3033, 334, 33, 4189, 4632, 322} sont à gauche du 0 et donc on peut dire que ces personnes consomment faiblement les denrées suivants : pain couscous pate riz fruit légume dessert et huile. Et 302 se trouvent à droite consomment en grande quatité les denrées "pain couscous pate riz fruit légume dessert et huile".
La majorité des individus se projettent au centre de gravité. Ce qui présume

que la plupart des individus consomment moyennement les denrées "pain couscous pate riz fruit légume dessert et huile".

3.1. Le deuxième axe factoriel

Le deuxième axe factoriel est expliqué par les variables suivantes : painjc, legumejc, laitjc, couscou1jc, patejc et legusecjc. Cet axe explique 9% de l'inertie totale.
C'est un axe d'opposition. Il oppose la consommation de couscou1jc, patejc et legusecjc à la consomation de painjc legumejc et laitjc. Autrement dit les personnes qui consomment beaucoup de pain, lait et légume s'opposent aux individus qui consomment beaucoup de legume sec, couscous et pate.
En ce qui concerne les personnes on remarque d'après le graphe ci dessous que l'individu 3909 consomme beaucoup de couscous, de legume sec et de pate ainsi que les individus 330, 4395, 4556, 1176.
En revanche les individus 1453, 1001, 3033, 326, 338, 1008, 3627, 334 consomment beaucoup de pain, legume et lait.

3.2. Le troisième axe factoriel

Cet axe explique $7,465\%$ de l'inertie totale.
C'est axe dopposition ; il oppose les variables painjc, pomterrjc aux variables fruitjc, viandejc, volailljc, oeufjc. Autrement dit les individus qui consomment beaucoup de painjc, pomterrjc consomment moins de fruitjc, viandejc, volailljc, oeufjc et inversement.

Projection des individus et les variables sur le premier plan factoriel

(voir annexe page 127)

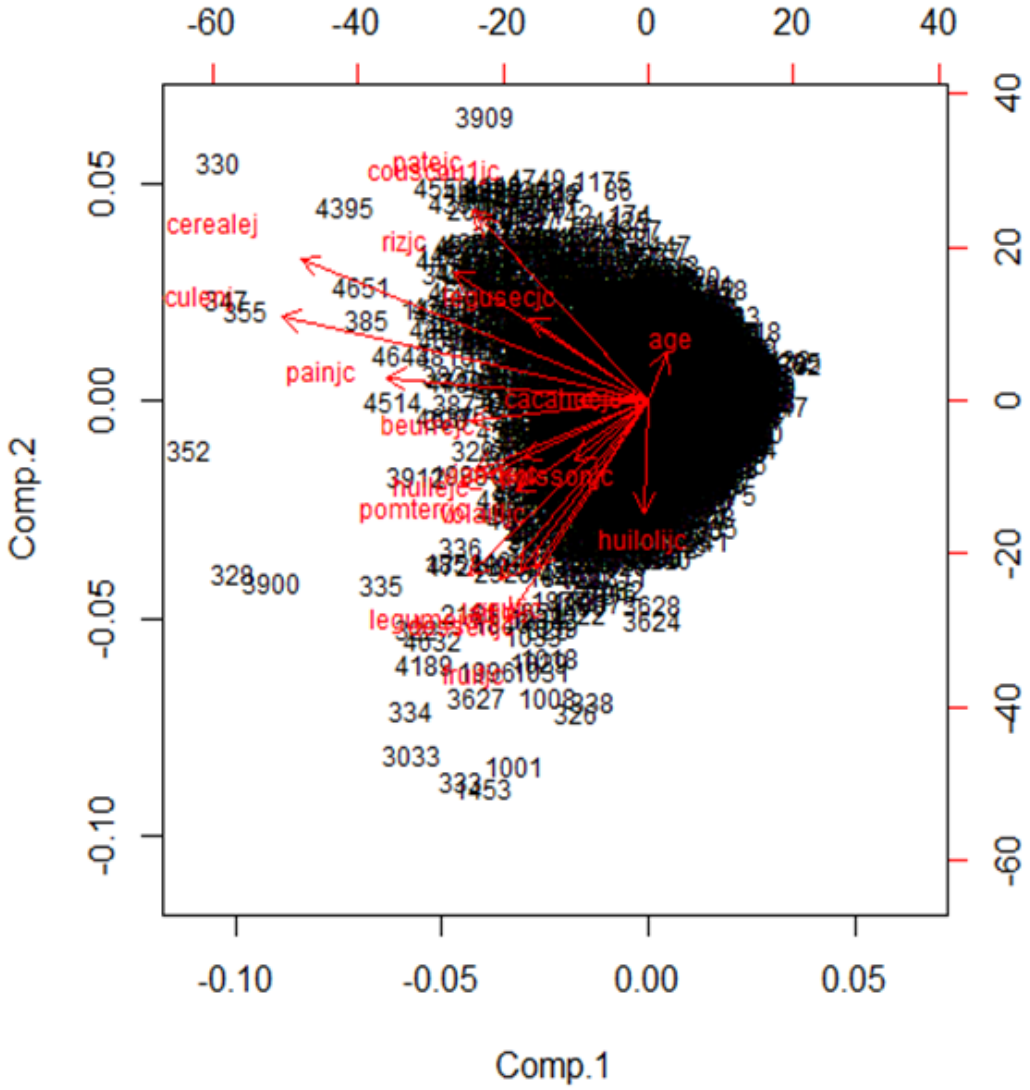

FIGURE V.2 – Projection des individus et les variables sur le premier plan factoriel

APPLICATION SUR RÉGRESSION LOGISTIQUE

INSP a mis à notre disposition une base de données constituée de 1312 variables sur un échantillon 4818 individus[1].

Toutefois, il y'a lieu de signaler que notre étude porte sur l'exploitation de base de données obtenue de TAHINA qui constitue l'outil de recueil d'informations qui permet de collecter grâce à un grand nombre de questions des renseignements relatifs à notre mode de vie, notre culture et nos habitudes alimentaires. A ce tire, seules 254 variables ont été retenues et ce après élimination des variables contenant plus de 10% de données manquantes.

A l'effet d'expliquer la teneur des données liées aux facteurs de risque d'atteinte du diabète, deux méthodes ont été utilisées à savoir la régression logistiques dichotomiques et la régression polytomique.(variables avec deux modalités et variables plus de deux modalités)

L'objectif de notre étude est d'expliquer et prédire une variable catégorielle Y (dichotomique) à partir d'une collection de descripteurs (covariable) $X = (X_1, X_2, .., X_J)$. Il s'agit en quelque sorte de mettre en évidence l'existence d'une liaison fonctionnelle sous-jacente de la forme :

$$Y = f(X, \alpha)$$

La fonction (de lien) f précise le modèle de prédiction α est le vecteur des paramètres de la fonction, on doit en estimer les valeurs à partir des données disponibles.
Dans le cadre de la discrimination binaire, nous considérons que la variable dépendante Y ne prend que 2 modalités : positif ou négatif. Nous cherchons à prédire les valeurs de Y, mais nous pouvons également vouloir quantifier la probabilité d'un individu à être positif (ou négatif).
Comme dans toute démarche de modélisation, plusieurs questions se posent immédiatement :
1. Choisir la forme de la fonction.
2. Estimer les paramètres du modèle à partir d'un échantillon.

3. Évaluer la précision des estimations.
4. Mesurer le pouvoir explicatif du modèle.
5. Vérifier s'il existe une liaison significative entre l'ensemble des descripteurs et la variable dépendante.
6. Identifier les descripteurs pertinents dans la prédiction de Y, évacuer celles qui ne sont pas significatives et/ou celles qui sont redondantes.
La régression logistique permet de répondre précisément à chacune de ces questions. Elle le fait surtout de manière complètement cohérente avec la maximisation de la vraisemblance.

Application la régression logistique pour des variables dichotomies(variables avec deux modalités)

.

groupe 1 :Ce groupe contient 21 variables explicatives.
Il contient des variables quantitatives concernant le cumul de consommation par semaine de certains alimentations :
« painjc, couscouljc, patejc, rizjc, legumejc, fruitjc, pomterrjc, "legusecjc, laitjc, poissonjc, viandejc, volailljc,oeufjc, cacahuejc, dessertjc, huilolijc, huilejc, beurrejc, feculenj, âge , cerealej».
La variable explicative étant le diabète prenant deux modalités qui sont diabete : sujet souffrant d'un diabète 1 : oui 2 non.

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.788e+00	3.599e-01	16.088	< 2e-16 ***
age	-6.755e-02	5.662e-03	-11.930	< 2e-16 ***
beurrejc	1.906e-01	1.285e-01	1.492	0.135729
cacahuejc	2.633e-01	2.576e-01	1.023	0.30641
cerealej	1.061e+05	1.464e+05	0.714	0.474999
coursesjc	-3.264e-04	828e-04	-0.478	0.632633
couscou1jc	1.234e+00	3.166e-01	3.934	8.34e-05 ***
dessertjc	6.815e-01	1.190e-01	5.728	1.02e-08 ***
feculenj	-1.046e+05	1.464e+05	-0.714	0.474998
fruitjc	1.158e-01	9.781e-02	1.184	0.236387
huilejc	5.637e-02	7.987e-02	0.706	0.480349
huilolijc	-6.160e-03	7.812e-02	-0.079	0.93715
laitjc	-4.092e-02	7.111e-02	-0.575	0.564989
legumejc	-2.578e-01	7.808e-02	-3.301	0.000963 ***
legusecjc	-4.381e-02	1.666e-01	-0.263	0.792538
oeufjc	4.658e-02	1.412e-01	0.330 0	0.741494
painjc	3.690e-01	2.205e-01	1.674	0.094179
patejc	9.597e-01	3.398e-01	2.825	0.004733 **
poissonjc	-3.614e-01	1.561e-01	-2.316	0.020567 *
pomterrjc	1.046e+05	1.464e+05	0.714	0.714
rizjc	-1.080e-01	3.501e-01	-0.309	0.757599
viandejc	-5.629e-01	1.847e-01	-3.048	0.002305 **
volailljc	-3.903e-01	1.990e-01	-1.961	0.049840 *

Nous remarquons que les 12 variables suivantes : beurrejc, cacahuejc, cerealej, feculenj, fruitjc ; huilejc, huilolijc, laitjc, legusecjc, pomterrjc, rizjc, oeufjc ne sont pas significatives. Il est à noter que les variables dans l'ensemble les matières grasses (beurrejc, cacahuejc, huilejc, huilolijc) et les féculents ne sont pas significatives dans l'explication de la maladie du diabète. La variable volailljc a une probabilité critique de 0.05080, elle est donc faiblement significative.

Nous procédons a une nouvelle estimation en retirant les variables dont le coefficient n'est pas significativement diffèrent de 0.

Nous refaisons la régression logistique pour les 10 variables significatives. Les résultats complets fournis par le logiciel R sont les suivants

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.795623	0.348864	16.613	< 2e-16 ***
age	-0.067587	0.005625	-12.016	< 2e-16 ***
couscou1jc	0.702314	0.185238	3.791	0.00015 ***
dessertjc	0.682888	0.115452	5.915 3	3.32e-09 ***
legumejc	-0.238479	0.074638	-3.195	0.00140 **
painjc	-0.076277	0.050002	-1.525	0.12714
patejc	0.366028	0.237112	1.544	0.12266
poissonjc	-0.365821	0.153807	-2.378	0.01739 *
viandejc	-0.547524	0.179478	-3.051	0.00228 **
volailljc	-0.3905	0.196130	-2.073	0.03817 *

Tel que le démontre la régression (2), la variable painjc, de même que patejc ne sont pas significatives et ont été ainsi retirées du modèle lors de cette étape. Après avoir éliminé à chaque fois les variables les moins significatives nous avons obtenu à la dernière étape le tableau suivant

Coefficients/	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.764272	0.335394	17.187	< 2e-16 ***
age	-0.068376	0.005607	-12.194	< 2e-16 ***
couscou1jc	0.809697	0.179316	4.515	2.13e-06 ***
dessertjc	0.712433	0.113790	6.245	4.23e-10 ***
legumejc	-0.203898	0.073400	-2.904	0.00368 **
poissonjc	-0.393896	0.150778	-2.709	0.00979 **
viandejc	-0.583747	0.172531	-2.947	0.00320 **
volailljc	-0.39337	0.18872	-2.084	0.03712 *

Interprétation des résultats :

a/ On devrait donc pouvoir mettre en place une régression qui permet d'estimer directement la probabilité d'avoir un diabétique P (Y = 1). En d'autres termes, le LOGIT estime permettant de prédire l'occurrence du diabète à partir de l ?âge et de la consommation de couscou1jc, dessertjc, legumejc, poissonjc, volailljc, viandejc s'écrit :

$C(X) = 5.894940 - 0.068594 * age + 0.852778 * couscou1jc + 0.712433 * dessertjc - 0.203898 * legumejc - 0.393896 * poissonjc - 0.393370 * volailljc - 0.522827 * viandejc$

Les résultats de cette régression sont concluants, du fait que les variables significatives dans ce modèle sont demeurées dans cette régression, malgré que certaines variables y aient été supprimées. Il est force de croire que l'âge de même que "dessertjc" ont un pouvoir prévisionnel dans la probabilité de d'apparition du diabète. On peut calculer le LOGIT pour chaque individu.

Exemple
pour (31) trente un individu avec les caractéristique suivantes :

couscou1jc	legumejc	poissonj	viandej	volaillj	dessertj	age	diabete
0 .2857143	2	0	0.285714	0	1	52.9936	1

$C(31) = 5.894940 - 0.068594 * 52.99384 + 0.852778 * 0.2857143 + 0.712433 * 1 - 0.203898 * 2 - 0.393896 * 0 - 0.393370 * 0 - 0.522827 * 0.2857143$

$C(31) = 2.65878$ $p = 1/(1 + exp(-2.6587891)) = 0.9345506 > 0.5$

Y = présence

La prédiction est correcte. En effet il s'agit de l'individu(no 31) dans notre tableau de données, il est positif ("présence").

b Signe du paramètre estimé

i) les variables (legumejc, poissonjc, viandejc , dessertjc , volailljc, âge) agissent négativement, les individus issus de personnes consommant ces denrées ont une présence de diabète plus faible.

ii)la variable (couscou1jc) est un facteur positif de diabète.

c Le odd ratio :
Il exprime un rapport de chances ; exemple odds = 2 l'individu à 2 fois plus de chances d'être positif que d ?être négatif. Le odd ratio indique le surcroît de chances d'être positif. Les personnes qui consomment du couscous ont presque 6 fois plus de chances (que les autres) d'avoir le diabète. On peut mesurer directement le surcroît de risque qu'introduit chaque facteur explicatif. Pour les variables quantitatives, le coefficient se lit comme consécutive à l'augmentation d ?une unité de la variable explicative. Résultats

Intercep	age	couscou	dessertj	legumejc	poissonjc	viandejc	volailljc
363.19	0.9337	2.3461	2.0389	0.81554	0.67442	0.59284	0.6748

Les résultats obtenus s'interprètent de la manière suivant : -L'odd ratio du variable « âge » est égal à 0.9337057 donc le risque d'avoir un diabétique lors d'une augmentation d'une unité du variable « âge » est multiplié par0.9337057.
- L'odd ratio de la variable « couscousjc » est égal à 2.346155 donc le risque d'avoir un diabétique lors d'une augmentation d'une unité de la variable couscousjc est multiplié par 2.346155

- L'odd ratio de la variable «dessertjc » est égal à 2.038946 donc le risque d'avoir un diabétique lors d'une augmentation d'une unité de la variable «dessertjc » est multiplié par 2.038946

- L'odd ratio de la variable «legumejc » est égal 0.7758625 donc le risque d'avoir une diabétique lors d'une augmentation d'une unité de la variable «legumejc » est multiplié par 0.8155456.

- L'odd ratio de la variable «poissonjc » est égal à 0.6744242donc le risque d'avoir une diabétique lors d'une augmentation d'une unité de la variable «poissonjc » est multiplié par 0.6744242.

- L'odd ratio de la variable «volailljc » est égal à 0.674779donc le risque d'avoir une diabétique lors d'une augmentation d'une unité de la variable « volailljc » est multiplié par 0.674779.
- L'odd ratio de la variable « viandejc » est égal à 0.5928422 donc une unité de satisfaction envers la variable « viandejc » augmente de 0.5928422 la probabilité d'avoir une diabétique.

Régression logistique du deuxième groupe : voir (annexe page 118)

Groupe2 : ce groupe contient 28 variables qualitatives « consommation de fruits » :

Les variables significatives sont : datte, pèche, pomme
Le modelé
$p(y = 1/x) = 2.9656 - 1.4519datte + 0.3019peche + 0.7258pomme$
Nous calculons le odd ratio :

Datte [T.1]	pêche [T.0]	Pomme [T.0]
0.234125	1.352426	2.066384

- Pour la variable datte :
Le risque d'avoir un diabétique chez un individu qui consomme les dattes est multiplié par 0.234125.
$_{P}ourlavariablep$êche :
Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pêches est multiplié par 1.352426.
- Pour la variable pomme :
Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pommes est multiplié par 2.066384.
Régression logistique du troisième groupe : voir (annexe)

Groupe 3 : groupe des légumes contient 35 variables qualitatives Les variables significatives sont : avocat, betterave, frites, saladver

Le modelé
$p(y = 1/x) = 1.0571 - 0.6395betterav - 0.2589frites + 0.4257saladver$
nous calculons le odd ratio

Avocat [T.1]	betterav[T.1]	frites[T.1]	saladver[T.0]
7.287059	0.5275561	0.7719002	1.530662

- Pour la variable avocat :
Le risque d'avoir un diabétique chez un individu qui consomme les avocats est multiplié par 7.287059.
- Pour la variable betterave :
Le risque d'avoir un diabétique chez un individu qui consomme les betteraves est multiplié par 0.5275561.
- Pour la variable frites : Le risque d'avoir un diabétique chez un individu qui consomme les frites est multiplié par0.7719002
- Pour la variable saladver : Le risque d'avoir un diabétique chez un individu qui ne consomme pas la salade vert est multiplié par 1.530662.
Régression logistique du deuxième groupe : voir (annexe)

Groupe 4 : ce groupe contient 14 variables concernant les produits laitiers.

Toutes les variables ne sont pas significatives.
Régression logistique du cinquième groupe : voir (annexe)
Groupe 5 : ce groupe contient 18 variables qualitatives des différents types des protéines (viandes et volailles et .) Les variables significatives sont : chameau, poisscon, mouton
Le modelé
$p(y = 1/x) = 5.8502 - 1.8895chameau - 1.8303poisscon + 0.3305mouton$
Nous calculons l' odd ratio

chameau[T.0]	poisscon[T.0]	mouton[T.1]
0.1511474	0.1603655	1.391664

- Pour la variable chameau :
Le risque d'avoir un diabétique chez un individu qui ne consomme pas la viande de chameau est multipliée par 0.1511474
- Pour la variable poisscon :
Le risque d'avoir un diabétique chez un individu qui ne consomme le poisson congelé est multiplié par 0.1603655.
- Pour la variable mouton :
Le risque d'avoir un diabétique chez un individu qui consomme la viande de mouton est multiplié par 1.391664.
Régression logistique du sixième groupe : voir (annexe)

Groupe 6 : ce groupe contient 23 variables qualitatives des légumes et fruits secs
Les variables significatives sont : viennois, dchicha

Le modelé
$p(y = 1/x) = 1.5168 + 0.4956viennois + 0.4842dchicha$
Nous calculons l'odd ratio

viennois [T.0]	dchicha[T.0]
1.641483	1.622876

-Pour la variable dchicha :
Le risque d'avoir un diabétique chez un individu qui ne consomme pas dchicha est

multiplié par 1.622876.
- Pour la variable viennois :
Le risque d'avoir un diabétique chez un individu qui ne consomme pas les viennois est multiplié par 1.641483.
Régression logistique du septième groupe voir (annexe)

Groupe7 : le groupe contient des variables qualitatives 17 concernant les différents types de gâteaux et les sucres.

Les variables significatives sont : confitur, patisser, sucre, sucrette

Le model
$p(y = 1/x) = 3.6988 - 1.0254confitur - 1.1749patisser - 1.3900sucre + 1.4063sucrette$
Nous calculons l'odd ratio

confitur[T.0]	patisser[T.0]	sucre[T.0]	sucrette[T.0]
0.358653	0.3088499	0.2490753	4.080828

- Le risque d'avoir un diabétique chez un individu qu'il ne consomme pas La confiture est multipliée par 0.358653. La pâtisser est multiplié par0.3088499 Le sucre est multiplié par 0.2490753 La sucrette est multipliée par 4.080828

Régression logistique du huitième groupe : voir (annexe)

Groupe 8 : contient des variables qualitatives et quantitatives concernant la description d'un individu.

Les variables significatives sont : actif, milieu, nbpièces, sexe

Le model $p(y = 1/x) = 3.96540 - 1.09793actif - 0.36966milieu - 0.09415nbpieces - 0.33059sexe$

Nous calculons l'odd ratio

actif[T.inactif]	milieu[T.urbain]	nbpieces	sexe[T.Hommes]
0.7498721	0.5913768	0.5235201	0.5819029

- Pour la variable actif :

Le risque d'avoir un diabétique chez un individu qu'il y inactif est multiplié par 0.7498721.

- Pour la variable milieu : Le risque d'avoir un diabétique chez un individu qu'il habite dans un milieu urbain est multiplié par 0.5913768.
- Pour la variable nbpieces :
Le risque d'avoir un diabétique chez un individu est liée par le nombre de pièces est multiplié par 0.5235201. - Pour la variable sexe :
Le risque d'avoir un diabétique chez un individu qu'il est homme est multiplié par 0.5819029.

Régression logistique du neuvième groupe : voir (annexe)

Groupe 9 : ce groupe contient des variables concernant le lieu et la convivialité des repas.

La variables significative est : exterier
Le model
$p(y = 1/x) = 2.59566 - 0.24223exterier$
Nous calculons l'odd ratio

exterieur[T.0]
0.5602631

Le risque d'avoir un diabétique chez un individu qu'il ne mange pas à extérieur est multiplié par 0.5602631.

Régression logistique du dixième groupe : voir (annexe)

Groupe 10 : contient des variables qualitatives concernant les boissons Les variables significatives sont : boissons, jusfruco, limonade, puits
Le model
$p(y = 1/x) = 4.937e + 00 - 2.168e - 04boissons - 1.300e + 00jusfruco - 7.810e - 01limonade - 7.522e - 01puits$
Nous calculons le odd ratio :

boissons	jusfruco	limonade	puits
0.9997832	0.2725318	0.4579478	1.580826

Pour la variable boisson :
Le risque d'avoir un diabétique chez un individu qui consomme

- Boissons gazeuses (boissons) est multipliée par 0.9997832.
  Le risque d'avoir un diabétique chez un individu qui ne consomme pas
- Jus de fruit commerce (jusfruco) est multipliée par 0.272531.
- Limonade est multipliée 0.4579478.
- Source (Puits) est multipliée 1.580826.

Régression logistique du onzième groupe : voir (annexe)

Groupe 11 : ce groupe contient des variables concernant les habitudes toxiques.

La variable significative est : tabfumsan

Le model
$P(y = 1/x) = 2.38216 + 0.46180tabfumsan$
Nous calculons L'odd ratio

tabfumsan[T.1]
0.3865589

la variable « tabfmsan1 » est égal à 0.3865589 donc les individus qui consomment du tabac sans fumer ont une probabilité de 0.3865589 d'avoir un diabétique.

Groupe 12 :ce groupe contient des variables concernant le loisir (passe-temps) les variables ne sont pas significatives

Le modèle final pour la variable diabète :

Il s'agit de faire la régression logistique avec les variables significatives retenues lors des régressions logistiques précédentes faites par groupe .

Les variables significatives sont : age, limonade, mouton, peche, sucre, sucrette

Model

$p(y = 1/x) = 5.467348age - 0.527523limonade + 0.325581mouton + 0.373315peche - 0.595898puits - 1.402585sucre + 1.263684sucrette$
Nous calculons L'odd ratio

age	limonade[T.0]	mouton[T.1]	pêche[T.0]	puits[T.0]	sucre[T.0]	sucrette[T.0]
0.9366374	1.384835	1.384835	1.452542	1.81466	-1.402585	3.538433

L'odd ratio du variable « âge » est égal 0.9366374 donc le risque d'avoir un diabétique lors d'une augmentation d'une unité du variable« âge » est multiplié par 0.9366374 .

Le risque d'avoir un diabétique chez un individu qui ne consomme pas

- Limonade est multipliée 1.38483.
- Source (Puits) est multipliée 1.81466

Pour la variable mouton : Le risque d'avoir un diabétique chez un individu qui consomme la viande de mouton est multiplié par 1.391664. Pour la variable pêche : Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pêches est multiplié par 1.452542. Le risque d'avoir un diabétique chez un individu qu'il ne consomme pas

- Le sucre est multiplié par 1.402585
- La sucrette est multiplié par 3.538433

la régression logistique polytomique

Régression logistique du huitième groupe : voir (annexe)

Pour groupe 8

A l'aide de package (vgam)sous R qui traite les variables de régression logistique polytomique Les variables significatives sont : situatm[T.divorcé] ,situatm[T.marié] , situatm[T.veuf] actiprin[T.retraité], actiprin[T.ouvrier], nivinst[T.moyen], nivinst[T.secondai] ,nivinst[T.supérieu] Les variables situatm[T.séparé], nivinst[T.primaire] ne sont pas significatives au seuil 0.05
Le model

$p(y = 1/x) = 3.57635 - 1.40338situatm[T.divorcé] - 1.02615situatm[T.marié] - 1.25124situatm[T.veuf] + 1.11805actiprin[T.ouvrier] - 0.71822actiprin[T.retraité] + 0.55382nivinst[T.moyen] + 1.07537nivinst[T.secondai] + 1.49377nivinst[T.supérieu]$

Nous calculons L'odd ratio

actiprin[ouvrier]	actiprin[retraité]	nivinst[moyen]	nivinst[T.secondai]	nivinst[T.supérieu]
0.2463731	0.6722149	0.3649786	0.2543832	0.1833565

situatm[T.divorcé]	situatm[T.marié]	situatm[T.séparé]	situatm[T.veuf]
0.8027197	0.7361688	0.8534787	0.7775144

Pour la variable activité principale (actiprin) le risque d'avoir un diabétique chez un individu qui

- ouvrier est multiplié par 0.2463731.
- retraité est multiplié par 0.6722149

Pour la variable niveau d'instruction (nivinst) Le risque d'avoir un diabétique chez un individu qu'il a un niveau d ?instruction

- Moyen est multiple par 0.3649786
- Secondaire est multiple par 0.2543832
- Supérieure est multiple par 0.1833565

Pour la variable situation mental (situatm) Le risque d'avoir un diabétique chez un individu qui

- Divorcé est multiplié par 0.8027197
- Marié est multiplié par 0.7361688
- séparé est multiplié par 0.8534787
- veuf est multiplé par 0.7775144

A l'aide de package (vgam) sous R qui traite la variable de régression logistique polytomique pour wilaya

les variables significative est wilaya [T.22 :sidi] les variables ne sont pas significatives : wilaya[T.7 :biskra] wilaya[T.5 :batna] wilaya[T.31 :oran] wilaya[T.16 :alger] au seuil 0.05 p(y=1/x)= 2.820055-0.806572wilaya[T.22 :sidi] Nous calculons L'odd ratio

wilaya[T.22 :sidi]
0.3086215

Pour la variable wilaya, le risque d'avoir un diabétique chez un individu qu'il habite à :

- Sidi Belabès est multiplié par 0.6722149.

Régression logistique du neuvième groupe : voir (annexe)

Groupe9

A l'aide de package (vgam)sous R qui traite la variable de régression logistique polytomique Les variables significatives sont : dejlieu ,dinavec Model
$P(y = 1/x) = 2.1697 + 0.7198dejlieu - 0.449dinavec$

Nous calculons L'odd ratio

dejlieu	dinavec
0.3274371	0.6104966

- Pour la variable dejlieu L'odd ratio de la variable dejlieu est égal à0.3274371désigne soit pour la restauration rapide ou chez vous ou bien au lieu de travail et restaurant donc le risque d'avoir une diabétique chez ces individus est multipliée par 0.3274371.
- Pour la variable dinavec

L'odd ratio de la variable « dinavec » est égal à 0.6104966 donc le risque d'avoir un diabétique chez les individus qui prennent leurs diners avec sa famille ou seul ou bien avec des amis est multiplié par 0.6104966.

Le modèle final pour la variable diabète selon les variables explicatives polytomiques.

A l'aide de package (vgam)sous R qui traite la variable de régression logistique polytomique les variables significatives sont : actiprin[T.ouvrier] ,actiprin[T.retraité] ,dinavec[T.2] nivinst[T.moyen] ,nivinst[T.secondai], nivinst[T.supérieu] ,situatm[T.divorcé], situatm[T.marié], situatm[T.veuf]

p(y=1/x)= 3.4816+1.1702 actiprin[T.ouvrier] -0.8396 actiprin[T.retraité] -0.5709 dinavec[T.2]+ 0.4875 nivinst[T.moyen] +0.9122 nivinst[T.secondai] +1.5337 nivinst[T.supérieu] -1.4483 situatm[T.divorcé] -1.1234 situatm[T.marié] -1.3422 situatm[T.veuf] les variables ne sont pas significatives situatm[T.séparé] ,actiprin[T.cadre su], nivinst[T.primaire] au seuil 0.05

actiprin[ouvrier]	actiprin[retraité]	nivinst[moyen]	nivinst[T.secondai]	nivinst[T.supérieu]
0.2463731	0.6722149	0.3649786	0.2543832	0.1833565

situatm[T.divorcé]	situatm[T.marié]	situatm[T.séparé]	situatm[T.veuf]
0.8027197	0.7361688	0.8534787	0.7775144

L'odd ratio de la variable « dinavec 2» est égal à 0.5650167 donc le risque d'avoir un diabétique chez les individus qui prennent leurs dine seul est multiplié par 0.5650167.
Pour la variable activite principle (actiprin) Le risque d'avoir un diabétique chez un individu qui est
- ouvrier est multiplié par 3.222637.
- retraité est multiplié par 0.4318832 Pour la variable niveau d ?instruction (nivinst)

Le risque d'avoir un diabétique chez un individu qu'il a un niveau d'instruction
- Moyen est multiplié par 1.628241
- Secondaire est multiplié par 2.489794
- Supérieur est multiplié par 2.489794 Pour la variable situation (situatm) Le risque d'avoir un diabétique chez un individu qui est
- Divorcé est multiplié par 0.2349694
- Marié est multiplié par 0.3251723
- séparé est multiplié par0.1800177
- veuf est multiplié par 0.2612702

Conclusion :

A partir des résultats obtenus, on peut aboutir à la conclusion suivante : On remarque qu'au départ nous avons 254 variables après la régression logistique dichotomie et polytomie, elles sont réduites à 40 variables significatives pour le diabète.

Les individus dont le régime alimentaire est basé sur les légumes et les fruits sauf (frite,avocat, betterav, datte)ont moins de risque d'avoir le diabète par rapport à ceux dont le régime alimentaire est basé sur la viande de mouton .

Les individus inactifs habitant dans un milieu urbain risquent d'avoir un diabète.

L'Age est un facteur de risque pour un diabétique.

APPLICATION SUR LES ARBRE DE DÉCISION

Application de la méthode RPART :

Objectif : prédire une variable en fonction d'attributs pour une liste d'individus. On suppose avoir une liste d'individus caractérisés par des variables explicatives, et on cherche à prédire une variable expliquée. L'apprentissage se fait par partitionnement récursif des instances selon des règles sur les variables explicatives. Deux types d'arbres de décision :

- **arbres de classification** : la variable expliquée est de type nominal (facteur). A chaque étape du partitionnement, on cherche à réduire l'impureté totale des deux nœuds fils par rapport au nœud père.
- **arbres de régression** : la variable expliquée est de type numérique et il s'agit de prédire une valeur la plus proche possible de la vraie valeur.

Nous avons appliqué la méthode RPART en utilisant les variables des deux modèles finaux de la régression logistique.

Dans notre cas nous utilisons les arbres de classification car la variable expliquée diabète est type nominal (facteur).

1-Application de la méthode RPART pour la variable diabète Nous avons appliqué la méthode RPART en utilisant le logiciel R3.0.3 en adoptant la méthodologie suivante :

- Diviser l'échantillon global en cinq sous échantillon aléatoire
- Appliquer sur chaque sous échantillon la méthode RPART.
- Calculer l'erreur de chaque sous arbre en utilisant la méthode leave (k-1) out.
- Choisir l'arbre qui présente le plus petit taux d'erreur.

application la méthode RPART sur les variables dichotomies

Nous avons appliqué la méthode RPART en utilisant les variables de premier modèle final de la régression logistique.

Notre base de données pour construire l'arbre de décision contient 4818 individus et 33 variables.

Pour la réalisation de chaque arbre associé au sous échantillon, on choisit 1000 individus aléatoires à l'aide de la fonction sample(fournit un échantillon aléatoire de 1000 valeurs prises dans le vecteur donné [1 :4818], mais avec remise.

actif ,age ,avocat ,betterav ,boissons, exterier ,frites ,jusfruco , legumejc, limonade, milieu ,mouton,nbpieces ,patisser ,peche ,poisscon, poissonjc ,pomme ,puits, saladver sexe ,sucre ,sucrette ,viandejc ,viennois,volailljc ,tabfumsan, couscou1jc, dessertjc, datte, chameau, milieu, dchicha, confitur .

Arbre associé au sous échantillon 1 : nous prenons les 1000 premiers individus aléatoires .

L'objet retourné est de la classe rpart. L'argument method="class" est optionnel (automatique par défaut car la variable prédite est de type facteur). Impression de l'arbre sous forme textuelle pour un arbre de classification : (Apprentissage) et 300 restants sont utilisés pour le test

A chaque noeud, on a :

La lecture des résultats obéit aux indications suivantes :

> numéro de noeud : 15 (noeuds de gauche et droite numérotés 2x et 2x + 1 si père numéroté x).
> le critère de split (ou root pour la racine) : age<40.3436 le nombre total d'instances pour le noeud : 139 le nombre d'instances mal classées : (0 => toutes les instances sont bien prédites) : 0 la valeur prédite (donc majoritaire) de la variable à prédire : 0 entre parenthèses, les proportions d'instances bien et mal prédites : (0.00000000 1.00000000) une '*' si c'est un noeud terminal.

n= 700

node), split, n, loss, yval, (yprob)
* denotes terminal node

1) root 700 50 0 (0.07142857 0.92857143)
2) age>=54.42847 220 35 0 (0.15909091 0.84090909)
4) sucre=0 64 23 0 (0.35937500 0.64062500)
8) confitur=0 56 23 0 (0.41071429 0.58928571)
16) puits=0 45 22 0 (0.48888889 0.51111111)
32) age>=66.54346 8 2 1 (0.75000000 0.25000000) *
33) age< 66.54346 37 16 0 (0.43243243 0.56756757)
66) age< 60.54346 21 9 1 (0.57142857 0.42857143)
132) boissons>=380 12 3 1 (0.75000000 0.25000000) *
133) boissons< 380 9 3 0 (0.33333333 0.66666667) *

67) age>=60.54346 16 4 0 (0.25000000 0.75000000) *
17) puits=1 11 1 0 (0.09090909 0.90909091) *
9) confitur=1 8 0 0 (0.00000000 1.00000000) *
5) sucre=1 156 12 0 (0.07692308 0.92307692) *
3) age< 54.42847 480 15 0 (0.03125000 0.96875000)
6) sucrette=1 7 2 0 (0.28571429 0.71428571) *
7) sucrette=0 473 13 0 (0.02748414 0.97251586)
14) age>=40.3436 334 13 0 (0.03892216 0.96107784)
28) age< 41.06913 14 3 0 (0.21428571 0.78571429) *
29) age>=41.06913 320 10 0 (0.03125000 0.96875000)
58) actif=inactif 200 10 0 (0.05000000 0.95000000)
116) volailljc>=0.5 11 3 0 (0.27272727 0.72727273) *
117) volailljc< 0.5 115 4 0 (0.03478261 0.96521739)
234) saladver=0 65 4 0 (0.06153846 0.93846154)
468) boissons< 475 46 4 0 (0.08695652 0.91304348)
936) boissons>=67.5 29 4 0 (0.13793103 0.86206897)
1872) legumejc>=0.5 8 1 0 (0.12500000 0.87500000) *
1873) legumejc< 0.5 8 0 0 (0.00000000 1.00000000) *
937) boissons< 67.5 17 0 0 (0.00000000 1.00000000) *
469) boissons>=475 19 0 0 (0.00000000 1.00000000) *
235) saladver=1 50 0 0 (0.00000000 1.00000000) *
59) actif=actif 117 0 0 (0.00000000 1.00000000) *
15) age< 40.3436 139 0 0 (0.00000000 1.00000000) *

Arbre de classification

Détermination de l'arbre optimal

La question se pose maintenant de déterminer quel est l'arbre optimal (celui qui risque le moins de faire des erreurs de prédiction). Par défaut, rpart() effectue un élagage de l'arbre et une validation croisée à 10 plis sur chaque arbre élagué. Les mesures effectuées au long de cette procédure sont stockées dans une table dénommée la cptable.

La commandead.datat cptable affiche cette table qui va nous permettre de répondre à la question précédente (quel est l'arbre optimal) :

Erreur de noeud racine : 50/700 = 0.071429

	CP	nsplit	rel error	xerror	xstd
1	0.016	0	1.00	1.00	0.13628
2	0.012	7	0.80	1.10	0.14238
3	0.010	16	0.68	1.16	0.14587

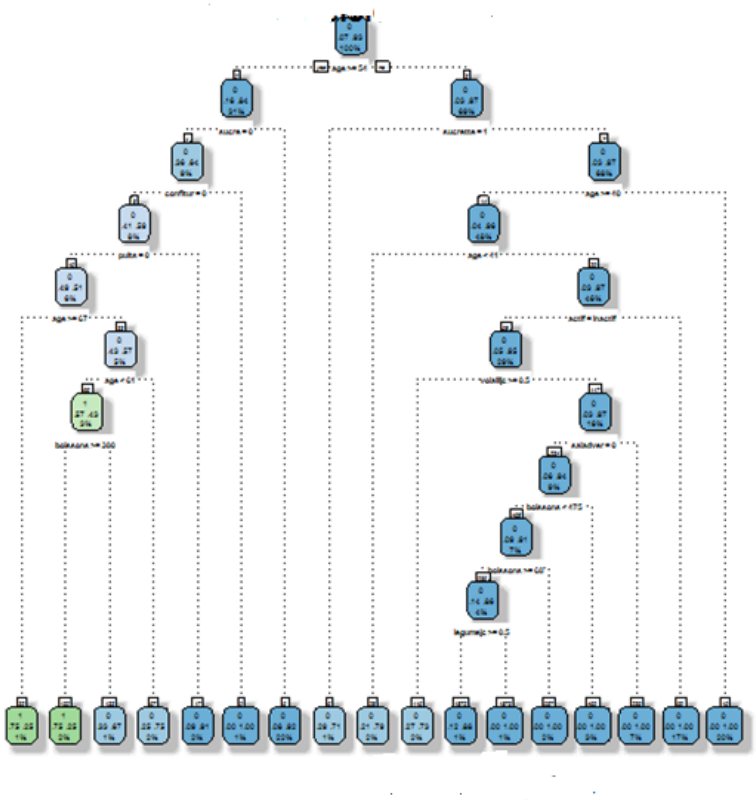

FIGURE V.3 – arbre de décision data x_1/*diabète*

Si on prend le premier nœud on a :
L'individu ayant plus de 54 ans, consommant du sucre et de la confiture risque d'être diabétique.
D'après ce chemin on constate que les trois facteurs la consommation du sucre(1) la consommation de la confiture(2) et l'âge d'un individu (3) sont des variables fortement corrélés avec le diabète. De plus l'âge de l'individu constitue un facteur de risque, en effet plus l'individu est âgé plus le risque de diabétique est élevé.

Si on prend le nœud (boissons) on constate que l'individu qui consomme des boissons <475 a plus de chance d'être non diabétique .

D'après les résultats obtenus par l'étude des données, la variable inactif apparait comme un facteur de risque, on remarque que les individus qui consomment les légumes sont moins touchés par le diabète.

Discussion :

Ces résultats obtenus sur l'étude de 1000 individus recensés en Algérie ont montré l'âge, le sucre et inactif sont fortement corrélés avec le risque du diabète.

On peut dire que les chances d'être non diabétique décroissent avec l'âge avancé des individus.

Les résultats obtenus par le logiciel R (Arbre de décision) sont confirmés par les médecins spécialistes.

Variables effectivement utilisés dans la construction de l'arbre

age	boissons	confitur	legumejc	sucre	puits	saladver	actif	sucrette	volailljc

Arbre associé au sous échantillon 2 :[annexe page]
nous prenons le deuxième 1000 individus aléatoires

Erreur de noeud racine :58/700 = 0.082857

	CP	nsplit	rel error	xerror	xstd
1	0.040230	0	1.00000	1.00000	0.12575
2	0.0013793	5	0.77586	1.2069	0.13685
3	0.010000	10	0.70690	1.2586	0.13942

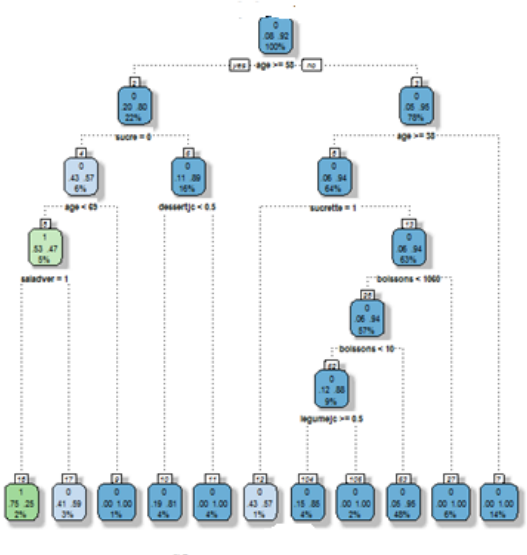

FIGURE V.4 – arbre de décision data x_2/*diabète*

Variables effectivement utilisés dans la construction de l'arbre

age	boissons	dessertjc	dessertjc	saladver	sucre	sucrette

Arbre associé au sous échantillon 3 :[annexe page]
nous prenons le deuxième 1000 individus aléatoires

Erreur de noeud racine : 58/700 = 0.082857

	CP	nsplit	rel error	xerror	xstd
1	0.071429	0	1.00000	1.00000	0.12817
2	0.017857	4	0.71429	0.94643	0.12498
3	0.010000	7	0.66071	0.98214	0.12712

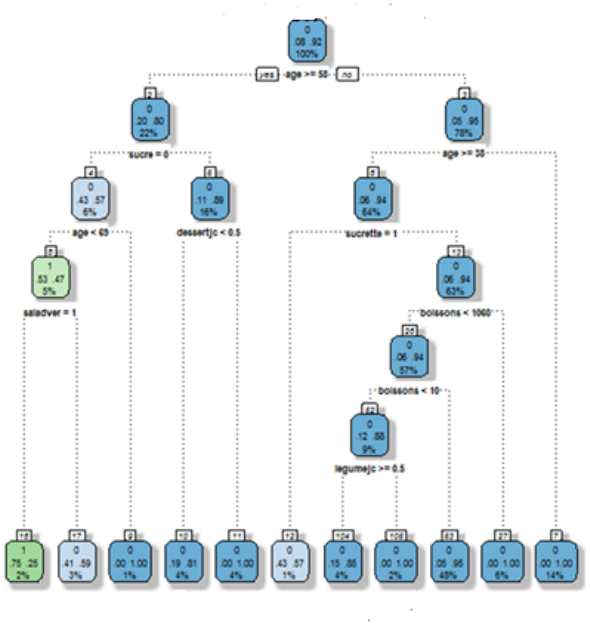

FIGURE V.5 – arbre de décision data x_3/*diabète*

Variables effectivement utilisés dans la construction de l'arbre

age	boissons	dessertjc	legumejc	sucre	sucrette	saladver

Arbre associé au sous échantillon 4 :[annexe page]

nous prenons le deuxième 1000 individus aléatoires

Erreur de noeud racine : $56/700 = 0.08$

	CP	nsplit	rel error	xerror	xstd
1	0.071429	0	1.00000	1.00000	0.12817
2	0.017857	4	0.71429	0.94643	0.12498
3	0.01000	7	0.66071	0.98214	0.12712

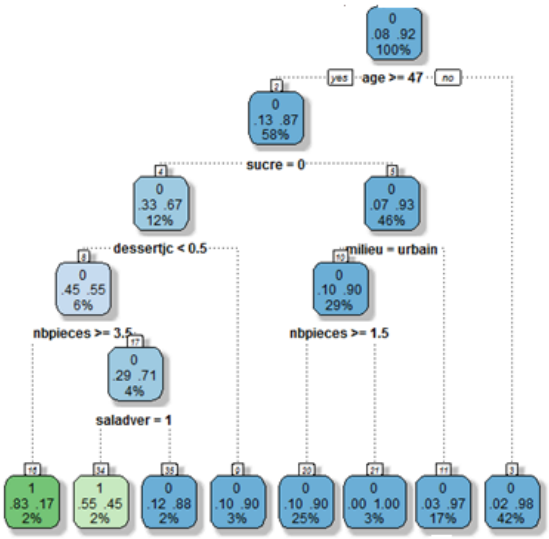

FIGURE V.6 – arbre de décision data x_4/*diabète*

Variables effectivement utilisés dans la construction de l'arbre

age	dessertjc	milieu	nbpieces	saladver	sucre

Arbre associé au sous échantillon 5 :[annexe page]
nous prenons les 1000 individus aléatoires

Erreur de noeud racine : $58/700 = 0.082857$

	CP	nsplit	rel error	xerror	xstd
1	0.071429	0	1.00000	1.00000	0.12817
2	0.017857	4	0.71429	0.94643	0.12498
3	0.010000	7	0.66071	0.98214	0.12712

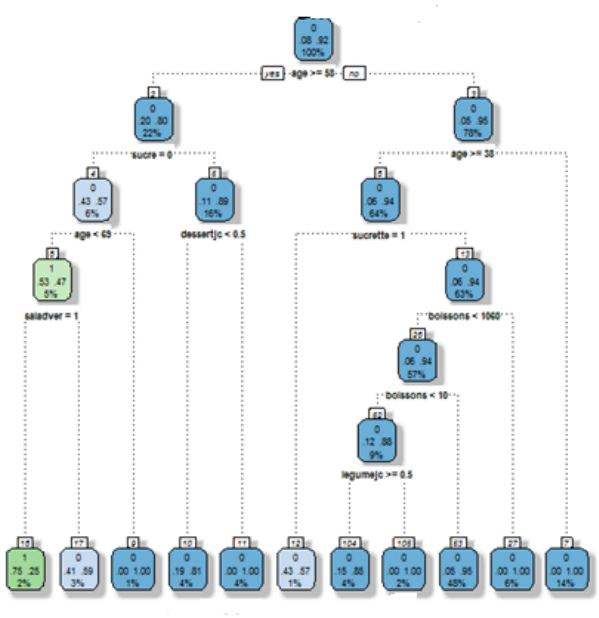

FIGURE V.7 – arbre de décision data x_5/*diabète*

Variables effectivement utilisés dans la construction de l'arbre

age	dessertjc	boissons	legumejc	saladver	sucre	sucrette

Validation par la méthode de leave (k-1) out :

Après avoir dérouler un programme sous R 3.0.3(package RPART) pour chaque arbres nous avons trouvées l'erreur associées à chaque arbre définie dans le tableau suivant :

Nechon	Erreur de noeud racine
1	$50/700 = 0.071429$
2	$58/700 = 0.082857$
3	$56/700 = 0.08$
4	$58/700 = 0.082857$
5	$58/700 = 0.082857$

L' arbre retenu est l'arbre $n01$ car il a le plus petit taux d'erreur.
Interprétation de l'arbre retenue :

diabète	0	1
Effectif	928	72

Construction de l'arbre complet

La formule utilisée diabete . Indique qu'on souhaite prédire la variable non diabète

en fonction de toutes les autres. Le principe général est que la (ou les) variable(s) à prédire sont à gauche du symbole alors que les variables prédictives sont à droite du symbole. Ici, le point. Permet d'indiquer qu'on souhaite utiliser toutes les variables des données comme prédicteurs sauf les variables à prédire (ce qui évite d'avoir à écrire la liste des prédicteurs).

On a utilisé ici les paramètres par défaut de la fonction rpart, ce qui ne conduit pas toujours à la solution désirée. En effet, rpart ne construit en général pas l'arbre le plus complet possible, pour des raisons d'efficacité. Il est rare en pratique qu'un arbre très profond qui ne réalise aucune erreur de classement sur les données d'apprentissage soit le plus adapté. Il sur-apprend massivement, en général. Il n'est donc pas très utile de construire un tel arbre, puisqu'on devra en pratique l'élaguer.

Cependant, il arrive sur des données de petite taille que les paramètres par défaut de rpart soient trop conservateurs. Par exemple, rpart ne découpe pas une feuille contenant 20 observations. De même rpart demande une amélioration relative d'au moins 1% de la qualité d'une partition pour effectuer un découpage. Pour changer ces valeurs, il suffit d'utiliser la commande rpart.control en précisant les éléments à modifier.

Construit un arbre en continuant les découpages dans les feuilles qui contiennent au moins 5 observations (paramètre minsplit) et sans contrainte sur la qualité du découpage (paramètre cp mis à 0). L'arbre construit de cette façon est assez volumineux et contient 16 feuilles.

Simplification de l'arbre
Niveaux de simplification
Pour choisir le bon niveau de simplification, ou encore le bon nombre de feuilles, on procède par validation croisée. La fonction rpart réalise par défaut une estimation des performances de l'arbre par validation croisée à 10 blocs pour chaque niveau de simplification pertinent. Le nombre de blocs se règle au moment de la construction de l'arbre grâce au paramètre xval de rpart.control.

On peut afficher les résultats de cette opération grâce à la fonction printcp, comme ci-dessous. La courbe indique le taux de mauvaises classifications relativement au score d'origine (dans un arbre réduit à une seule feuille dans laquelle la décision correspond à la classe majoritaire), estimé par la validation croisée. Les barres d'erreur autour de chaque estimation sont aussi obtenues par validation croisée. Ici, comme on a 7 diabétiques et 927 non diabétique, l'erreur de référence est d'environ 92.8%. L'axe des abscisses indique la complexité de l'arbre par l'intermédiaire du nombre de feuilles.

Performances par validation croisée

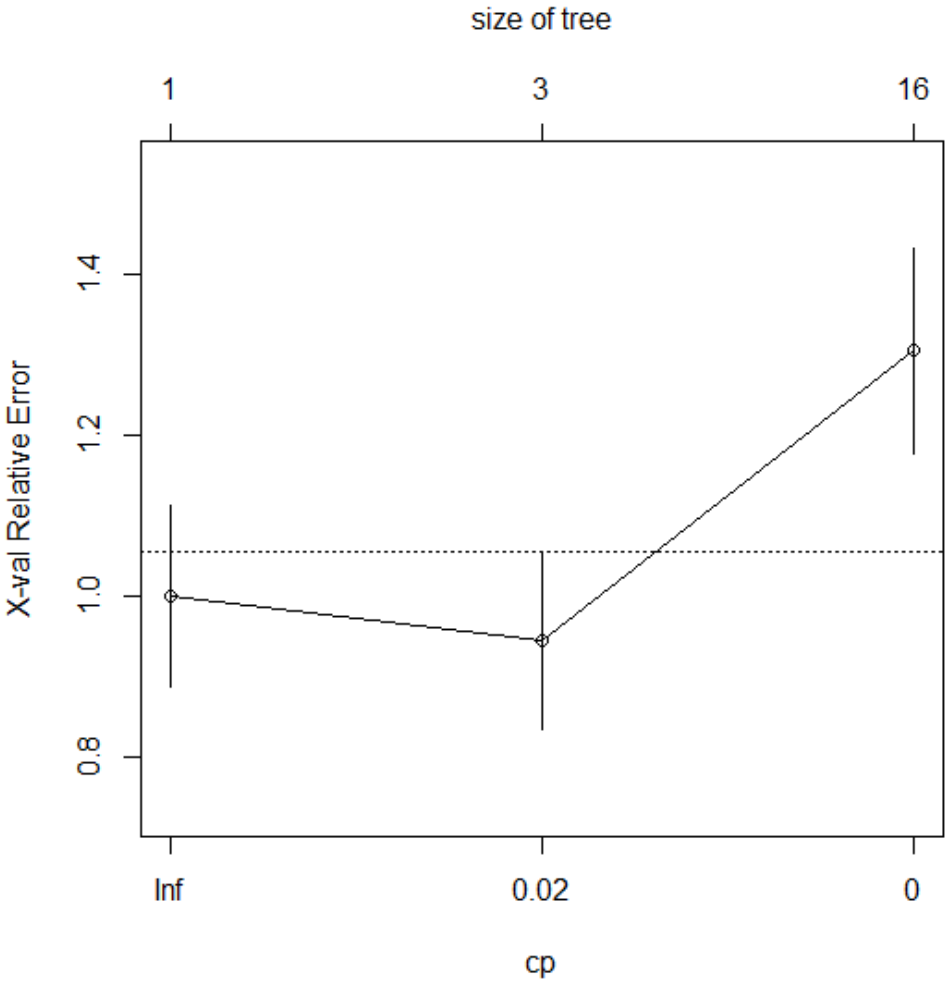

FIGURE V.8 – Performances par validation croisée

Simplification

Comme attendu, les performances s'améliorent dans un premier temps quand on augmente le nombre de feuilles puis se dégradent en raison du sur-apprentissage. On choisit en général la complexité qui minimise l'erreur estimée, soit ici 3 feuilles

On a utilisé ici les paramètres par défaut de la fonction rpart, ce qui ne conduit pas toujours à la solution désirée. En effet, rpart ne construit en général pas l'arbre le plus complet possible, pour des raisons d'efficacité. Il est rare en pratique qu'un arbre très profond qui ne réalise aucune erreur de classement sur les données d'apprentissage soit le plus adapté. Il sur-apprend massivement, en général. Il n'est donc pas très utile de construire un tel arbre, puisqu'on devra en pratique l'élaguer.

Évaluation des performances

Pour évaluer correctement les performances de l'arbre simplifié, il faut utiliser une procédure de type validation croisée en plus de celle qui est intégrée dans rpart pour choisir la complexité de l'arbre. Le package caret facilite cette opération en propo-

sant des fonctions de découpage des données bien conçues.

Dans cette prise en main, nous nous contentons d'évaluer les performances sur les données d'apprentissage ce qui sur-estime la qualité du modèle. Il s'agit simplement d'une illustration ! Le principe est de s'appuyer sur la fonction predict et sur la fonction table pour obtenir :

une matrice de confusion

	Diabétique(1)	Non diabétique (0)
Diabétique(1)	7	65
Non diabétique (0)	1	927

On constate que la qualité de la prédiction dépend beaucoup de la classe. En effet, sur les 7 diabétiques, le taux de prévisions correctes est de $9,72\%$ environ, alors que sur les 927 non diabétique, il est de 99.89%.
Arbre associé à l'échantillon global :

on applique la méthode RPART sur l'echantillon global
Nous avons appliqué la méthode RPART en utilisant les variables dichotmies sur l'echanlillon global 4818 individus .

Erreur de noeud racine :264/3372 = 0.078292.

	CP	nsplit	rel error	xerror	xstd
1	0.037879	0	1.00000	1.00000	0.059087
2	0.015152	4	0.84848	1.0417	0.060199
3	0.011364	9	0.77273	1.0379	0.060099
4	0.010417	16	0.66667	1.0795	0.061185
5	0.010000	20	0.62500	1.0720	0.060989

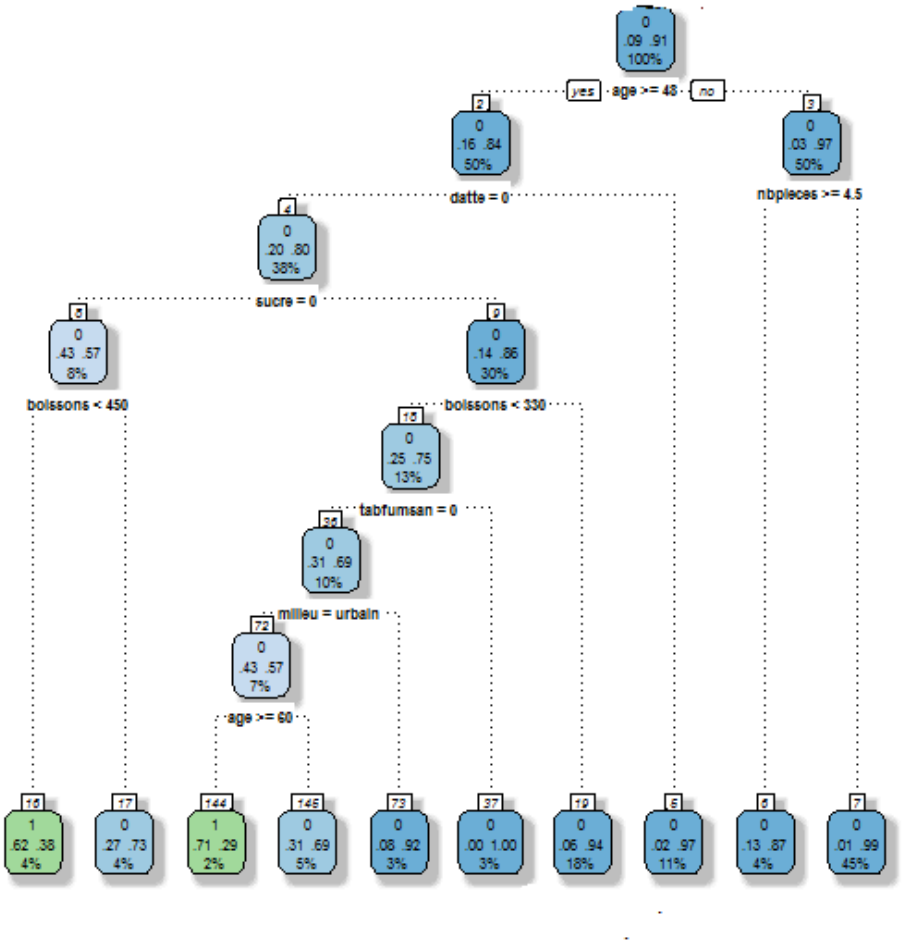

FIGURE V.9 – Arbre associé à l'échantillon global

Discussion :

Ces résultats obtenus sur l'étude de 4818 individus recensés en Algérie ont montré l'âge, le sucre et boissons sont fortement corrélés avec le risque du diabète.

On peut dire que les chances d'être non diabétique décroissent avec la consommation les légumes ,la salade vert et les fruits comme (les péchés et les pommes) chez des individus.

Variables effectivement utilisés dans la construction de l'arbre

age	boissons	confitur	couscou1jc	dessertjc	legumejc	limonade

milieu	mouton	peche	pomme	saladver	sucre

application la méthode RPART sur les variables polytomies

Nous avons appliqué la méthode RPART en utilisant les variables e deux modèle final de la régression logistique.

on applique la méthode RPART sur l'echantillon
Erreur de noeud racine :53.194/700 = 0.075992.

	CP	nsplit	rel error	xerror	xstd
1	0.028471	0	1.00000	1.0029	0.11470
2	0.013436	2	0.94306	1.0299	0.11277
3	0.011209	3	0.92962	1.0239	0.11070
4	0.010000	6	0.89599	1.0359	0.11158

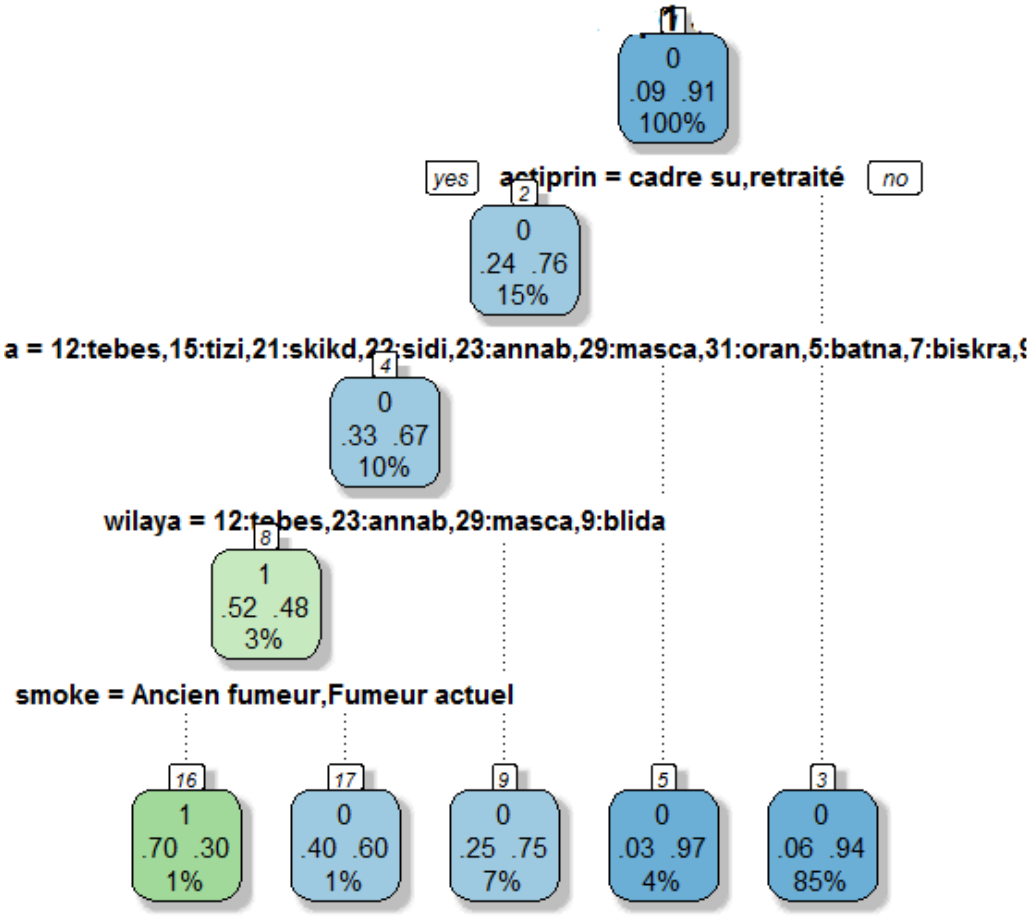

FIGURE V.10 – Arbre de décision datap/diabète

Variables effectivement utilisés dans la construction de l'arbre

actiprin	nivinst	situatm	wilaya

Conclusion :

Les résultats sont conformes aux résultats émis par les spécialistes en diabétologie, basés sur d'autres méthodes statistiques :

- l'âge est un facteur prédictif du diabète.
- Les individus dont le régime alimentaire est basé sur les légumes ont moins de risque d'avoir un diabétique.
- Les individus dont le régime alimentaire est basé sur les légumes ont moins de risque d'avoir un diabétique
- Les individus qui ne consomment pas de sucre sont susceptibles d'être touchés par le du diabète.
- Les individus inactifs sont prédisposés au diabète.
- Il y'a lieu de noter par ailleurs que les résultats obtenus sur 1000 individus sont identiques à ceux obtenus lors de l'étude des 4818 individus.

APPLICATION SUR RÉSEAUX DE NEURONES

Notre étude est rendue facile par le très faible nombre de descripteurs (p = 2) qui permet de projeter les observations dans le plan et, ainsi, d'identifier le bon paramétrage. En pratique, cette démarche n ?est pas tenable, surtout lorsque le nombre de descripteurs augmente (p > 2). Il nous définir une stratégie générique qui permet d'identifier automatiquement le bon nombre de neurones sur un jeu de données. R est un outil privilégié pour cela. Tout simplement parce qu'il est très facile de programmer des actions complexes avec quelques lignes de codes bien senties.

Dans cette section, dans un premier temps, nous reproduisons l'étude ci-dessus (sur « data ») pour décrire la mise en oeuvre du perceptron multicouche avec le package nnet. Puis, dans un second temps, nous mettons en place un dispositif pour spécifier le nombre adéquat de neurones de la couche cachée pour le fichier « data » où le concept à apprendre est plus complexe c.-à-d. il faut plus de droites séparatrices, mais combien ?

Les données

Nous partitionnons aléatoirement les données en deux échantillons de taille égale : la première, dite d ?apprentissage, servira à la construction du réseau ; la seconde, dite de test, sera utilisée pour évaluer les performances en généralisation.

Nous constatons que les caractéristiques des deux échantillons sont très proches. C'est heureux, la subdivision ayant été réalisée de manière aléatoire. Il faudrait s'inquiéter dans le cas contraire

Construction du modèle

Nous utilisons le package « nnet » dans ce tutoriel. Il présente l'avantage d'être particulièrement simple à utiliser. Ses sorties sont en revanche très sobres, voire laconiques. Nous pouvons quand même obtenir les poids synaptiques du perceptron.

La commande nnet() construit le réseau à partir de l'échantillon d'apprentissage. Nous fixons le nombre d'itérations maximum à 300 (maxit = 300) pour nous assurer de la stabilité des résultats. La valeur par défaut (maxit = 100) n'est apparemment pas suffisante pour nos données. L'option « skip = FALSE » indique la présence d'une couche cachée, « size = 2 » définit le nombre de neurones.

Pour le modelé dichotomie

Nous obtenus les résultats suivantes :

la fonction a approché étant une fonction à deux valeurs 0,1 sur son domaine de définition ,donc elle n' est pas régulière .les résultat obtenus par les réseaux multicouches s'interprète comme une approximation de cette fonction ,qui prend des valeurs proche de0 et de 1 sur le domaine de définition de la fonction initial ,elle peut être interpréter comme une fonction discriminante de $(0, 1)$ des deux classes(diabétique, non diabétique)

les poids$w_{i.j}$

b->h1	i1->h1	i2->h1	i3->h1	i4->h1	i5->h1	i6->h1	i7->h1	i8->h1	i9->h
0.11	4.50	-0.04	0.41	-0.51	-0.36	-0.26	-0.30	0.24	-0.02
i10->h1	i11->h1	i12->h1	i13->h1	i14->h1	i15->h1	i16->h1	i17->h1	i18->h1	i19->h
0.31	0.20	-0.44	0.20	-0.15	0.00	-0.55	-0.23	-0.04	0.55
i20->h1	i21->h1	i22->h1	i23->h1	i24->h1	i25->h1	i26->h1	i27->h1	i28->h1	i29->h
0.61	0.26	0.39	-0.10	-0.07	0.38	0.73	-0.34	0.45	-0.18

i30->h1	i31->h1	i32->h1	i33->h1
24.26	-0.53	-0.36	0.62

les poids$w_{j.k}$

b->h2	i1->h2	i2->2	i3->h2	i4->h2	i5->h2	i6->h2	i7->h2	i8->h2	i9->h2
-0.11	0.35	0.45	0.64	0.26	0.00	-0.31	-0.38	-0.68	0.32
i10->h2	i11->h2	i12->h2	i13->h2	i14->h2	i15->h2	i16->h2	i17->h2	i18->h2	i19->h
-0.35	-0.47	-0.68	-0.02	-0.56	0.42	-0.20	0.61	-0.36	-0.04
i20->h2	i21->h2	i22->h2	i23->h2	i24->2	i25->h2	i26->h2	i27->h2	i28->h2	i29->h
-0.43	0.12	-0.06	-0.05	-0.14	0.01	-0.66	-0.54	-0.04	-0.14

i30->h2	i31->h2	i32->h2	i33->h2
0.47	0.37	0.10	-0.07

b->o	h1->o	h2->o
-12.56	-3.75	-12.57

La variable cible **(diabète)** étant binaire, de fait, R n'en décrit qu'une seule. Nous obtenons le réseau suivant :

les résultats obtenus dans notre travail montre que les réseaux multicouches à une seule couche cachée convient à notre modélisation .De plus en changent le nombre de neurones d'entrée et le nombres d'itération l'erreur n'a pas diminué. Ainsi pour notre travail plusieurs simulations ont été effectuées ,conduisant au fait qu'un réseau à 13 neurones dans la couche cachée suffisant à la modélisation du diagnostic d'un malade comme le diabète

Évaluation

Nous élaborons une fonction pour l'évaluation. Elle prend en entrée les données à utiliser et le modèle. Ce dernier se charge de produire la prédiction à l'aide de la commande predict(). Nous construisons la matrice de confusion à partir de la confrontation entre la cible observée et la prédiction. Nous en déduisons le taux d'erreur (taux de mauvais classement).

Appliquée sur l'échantillon test (2409 observations) et le réseau construit dans la section précédente,

Nous obtenons un taux d'erreur de 0.9379845.

Algorithme pour le paramétrage automatique du réseau

Notre objectif dans cette section est de mettre en place une procédure très simple de détection du nombre « optimal » de neurones de la couche cachée. Elle doit être générique c.-à-d. s'appliquer quel que soit le nombre de variables prédictives. Nous utiliserons alors pour apprendre la fonction ? le concept - associant (les variables) à diabète pour les données « data1 ».

Algorithme de paramétrage automatique

La démarche est très similaire à la stratégie « wrapper » utilisée pour la sélection de variables en apprentissage supervisé. Les données sont subdivisées en 2 échantillons : apprentissage et validation. Nous construisons différentes hypothèses sur l'échantillon d'apprentissage, dans notre cas il s'agit de nombre de neurones différents dans la couche cachée (1, ce qui revient à un perceptron simple ; puis 2, puis 3, etc.), que nous évaluons sur l'échantillon test. La solution correspond à la configuration la plus simple, c.-à-d. avec le plus petit nombre de neurones, permettant de minimiser le taux d'erreur en généralisation. Le programme tient en une boucle passant en revue différentes valeurs de $\acute{}k\check{}$, nombre de neurones dans la couche cachée. Nous avons fixé ex ante le nombre maximal de neurones à tester ($K = 20$). C'était le plus simple à implémenter. Mais nous pouvons également imaginer des stratagèmes plus élaborés (ex. s'arrêter dès que le taux d'erreur ne décroît plus). Le nom de la variable cible doit être diabète dans ce petit programme. En revanche, il n'y a aucune limitation concernant le nombre et le nom des prédicateurs.

Application sur les données « data1»

Pour vérifier l'efficacité du programme, nous l'appliquons sur les données « data1 » dont nous ne sommes pas censés connaître le nombre adéquat de neurones . Nous montrons ici le processus complet : l'importation des données, leur subdivision en apprentissage-validation, la mise en œuvre de la détection, l'affichage des résultats (nombre de neurones vs. taux d'erreur en validation) dans un graphique.

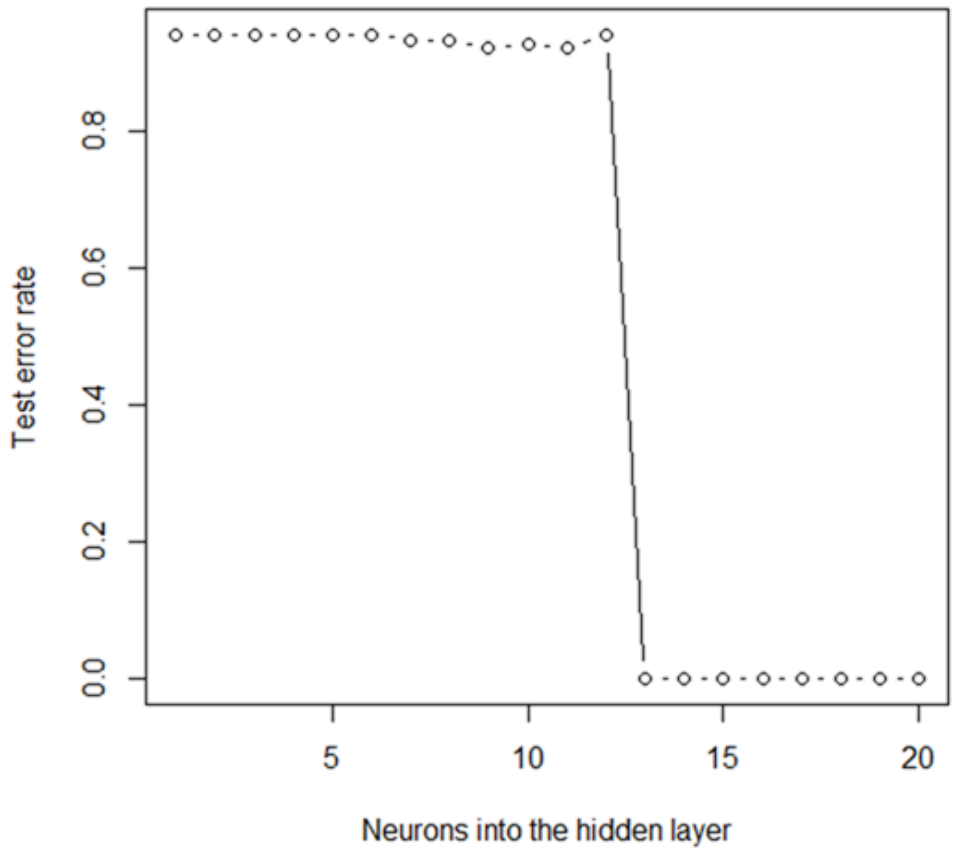

FIGURE V.11 – la courbe

Nous obtenons une courbe particulièrement édifiante. A partir de « k = 13 » neurones dans la couche intermédiaire, nous reproduisons parfaitement le concept. les résultats obtenus dans notre travail montre que les réseaux multicouches à une seule couche cachée convient à notre modélisation .De plus en changent le nombre de neurones d'entrée et le nombres d'itération l'erreur n'a pas diminué. Ainsi pour notre travail plusieurs simulations ont été effectuées ,conduisant au fait qu'un réseau à 13 neurones dans la couche cachée suffisant à la modélisation du diagnostic d'un malade comme le diabète.

Pour modelé polytomique :

les poids $w_{i.j}$

b->h1	i1->h1	i2->h1	i3->h1	i4->h1	i5->h1	i6->h1	i7->h1	i8->h1	i9->h
-14.62	-0.51	-2.07	0.16	-3.00	-1.33	-0.68	-0.67	-9.99	-0.10
i10->h1	i11->h1	i12->h1	i13->h1	i14->h1	i15->h1	i16->h1	i17->h1	i18->h1	i19->h
0.14	-3.56	-0.87	-0.01	-0.79	-0.47	-5.12	-0.34	-1.39	0.31
i20->h1	i21->h1	i22->h1	i23->h1	i24->h1	i25->h1	i26->h1	i27->h1	i28->h1	i29->h
0.61	0.26	0.39	-0.10	-0.07	0.38	0.73	-0.34	0.45	-0.18
i30->h1	i31->h1	i32->h1	i33->h1	i34->h1	i35->h1	i36->h1	i37->h1	i38->h1	i39->h
-1.02	-1.16	-0.69	-0.36	-1.13	-0.75	-0.55	-1.25	0.23	-0.42

i40->h1	i41->h1	i42->h1	i43->h1
-0.36	-0.45	-1.96	-10.00

les poids$w_{j.k}$

b->h2	i1->h2	i2->2	i3->h2	i4->h2	i5->h2	i6->h2	i7->h2	i8->h2	i9->h2
-13.09	-0.64	-1.87	-0.13	-3.05	-0.57	-0.47	0.31	-9.51	-0.04
i10->h2	i11->h2	i12->h2	i13->h2	i14->h2	i15->h2	i16->h2	i17->h2	i18->h2	i19->h
-0.47	-2.96	-0.45	-0.20	-0.56	-0.31	-5.58	-0.54	-0.61	-0.22
i20->h2	i21->h2	i22->h2	i23->h2	i24->2	i25->h2	i26->h2	i27->h2	i28->h2	i29->h
-3.85	-0.70	-0.38	-0.67	-2.01	-1.30	-1.12	-1.38	-0.76	-0.74
i30->h1	i31->h1	i32->h1	i33->h1	i34->h1	i35->h1	i36->h1	i37->h1	i38->h1	i39->h
-1.32	-0.97	-0.28	-0.12	-1.43	-1.25	-0.58	-1.40	-0.39	-0.04

i40->h2	i41->h2	i42->h2	i43->h2
-0.42	-0.73	-0.88	-9.40

b->o	h1->o	h2->o
2.44	20.05	3.35

La variable cible (diabète) étant binaire, de fait, R n'en décrit qu'une seule. Nous obtenons le réseau suivant :
Appliquée sur l'échantillon test (2409 observations) et le réseau construit dans la section précédente :
Nous obtenons un taux d'erreur de 0.9214317.

Algorithme pour le paramétrage automatique du réseau :
Application sur les données « data2»

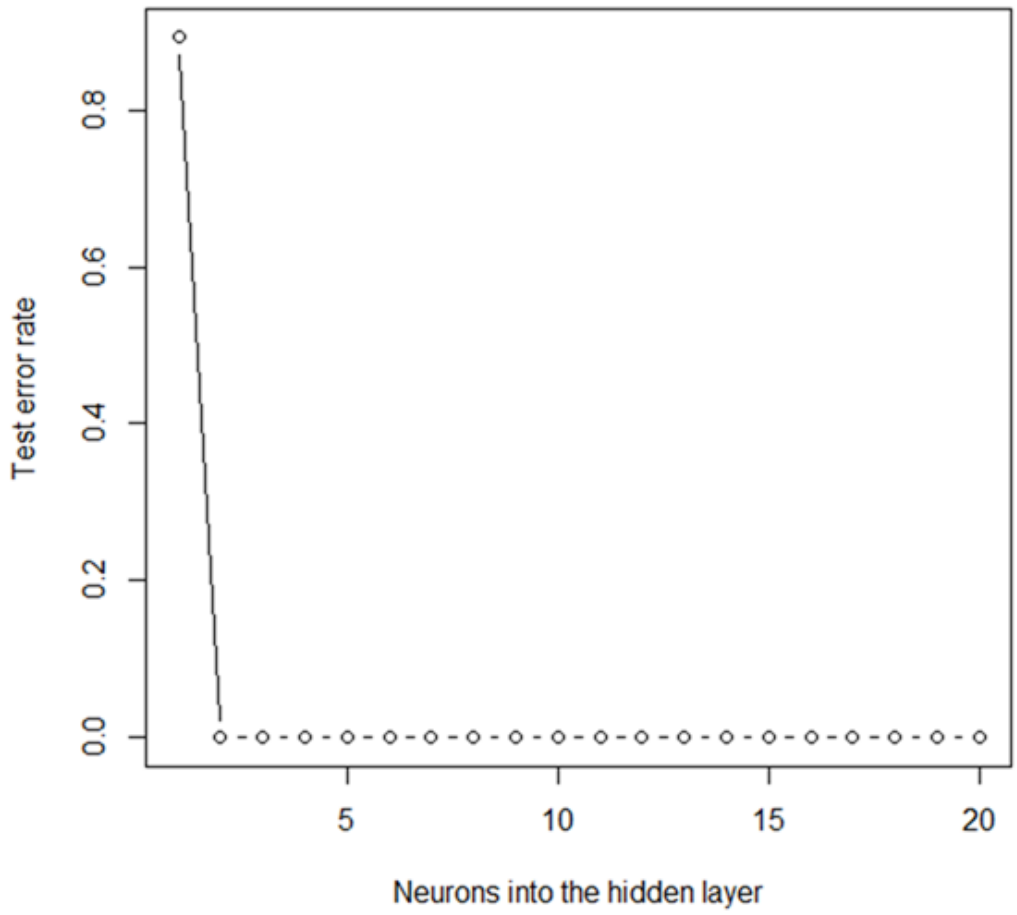

FIGURE V.12 – la courbe

Nous obtenons une courbe particulièrement édifiante. A partir de « k = 2 » neurones dans la couche intermédiaire, nous reproduisons parfaitement le concept. les résultats obtenus dans notre travail montre que les réseaux multicouches à une seule couche cachée convient à notre modélisation .De plus en changent le nombre de neurones d'entrée et le nombres d'itération l'erreur n'a pas diminué. Ainsi pour notre travail plusieurs simulations ont été effectuées ,conduisant au fait qu'un réseau à 02 neurones dans la couche cachée suffisant à la modélisation du diagnostic d'un malade comme le diabète.

CHAPITRE VI

IMPACT DU DIABÈTE SUR LES AUTRES MALADIES CHRONIQUES

Introduction

: Après avoir étudié les causes du dans les chapitres précédents, nous allons étudie brièvement la relation qui existe entre le diabète et certaine maladies chroniques comme : l'HTA « hypertension artérielle », l'obésité et les pathologies cardiovasculaires puis faire des prévisions sur ses maladies thème qui intéresse l'INSP.

Les variables utilisés comme des variables indépendantes sont : âge, sexe, l'IMC ,obèse , selon l'IDF, l'IMC, triglycéride, glycémie, cholestérol, l'antécédent familial de chaque maladie. Pour cela nous allons rappeler quelque notion concernant les mesures cliniques et la définition de quelque variable qui vont être utilisé dans cette partie.

Répartition selon les classes de glycémie :

- Glycémie normale : glycémie inférieure a 110 mg/dl.
- Glycémie modérée à jeun : glycémie comprise entre 110 mg/dl et 125 mg/dl.
- Hyperglycémie : glycémie supérieure ou égale a 126 m/dl.

Répartition selon les classes de cholestérolémie :

- Cholestérolémie normal : cholestérol inférieur a 200 mg/dl.
- Cholestérolémie limite : cholestérol compris entre 200 et 249 mg/ dl.
- Hypercholestérolémie : cholestérol supérieur ou égal a 250 mg/ dl.

Répartition selon les classes de triglyceridemie :

- Triglyceridemie normal : inférieure a 150 mg/dl.
- Triglyceridemie limite : compris entre 150mg/dl et 199mg/dl.
- Hypermg,yeender,e : supérieure ou égal a 200 mg/d,.

Répartition selon imc(Indice de masse corporelle)
on applique les trois méthode

régression logistique

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.800631	2.796206	0.286	0.77463
imc	0.050119	0.056919	0.881	0.037857**
maigreur[T.1]	0.160546	0.682613	0.235	0.81406
surpoids[T.1]	-0.190784	0.343216	-0.556	0.57830
obese[T.1]	0.711528	0.389760	1.826	0.06792 .
hypten[T.1]	0.333433	0.345886	0.964	0.33505
hypgly[T.1]	-1.647592	0.309564	-5.322	1.02e-07 ***
hyptri[T.1]	-0.223401	0.397074	-0.563	0.57369
hypgl[T.1]	-1.759153	0.281824	-6.242	4.32e-10 ***
IDF[T.1]	-0.291412	0.551413	-0.528	0.59716
ATPIII[T.1]	0.606668	0.473933	1.280	0.20052
IDFchol[T.1]	0.689721	0.530280	1.301	0.19337
ATPIIIchol[T.1]	-0.702193	0.429879	-1.633	0.10237
nouvhta[T.1]	-0.779954	0.466232	-1.673	0.09435 .
abdobesATP[T.1]	-0.401465	0.325139	-1.235	0.21692
abdobesIDF[T.1]	0.255315	0.369971	0.690	0.49014
tourtail	-0.026777	0.012349	-2.168	0.03013 *
poids	0.006435	0.013905	0.463	0.64354
glycemie	-0.007710	0.002552	-3.021	0.00252 **
cholest	-0.002324	0.002633	-0.882	0.37759
triglyce	0.001450	0.002356	0.616	0.53817
insufren[T.1]	0.965644	1.600477	0.603	0.54628
ulceregd[T.0]	-0.489123	0.650903	-0.751	0.45238
depress[T.0]	0.539397	0.733721	0.735	0.46225
cancer[T.0]	-0.381449	1.422706	-0.268	0.78861
asthme[T.0]	0.568924	0.438863	1.296	0.19485
pathcv[T.0]	0.001114	0.398954	0.003	0.99777
dyslipid[T.0]	1.608515	0.354598	4.536	5.73e-06 ***
hta[T.0]	1.760326	0.270876	6.499	8.10e-11 ***
antediab[T.0]	1.470038	0.202889	7.246	4.31e-13 ***
antehta[T.0]	-0.256083	0.201791	-1.269	0.20442

Après avoir éliminé les variables ayons un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau si dessus :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.173357	0.588335	0.295	0.76826
antediab[T.0]	1.415309	0.145761	9.710	< 2e-16 ***
ATPIIIchol[T.0]	-0.218952	0.178553	-1.226	0.22010
dyslipid[T.0]	1.580182	0.268024	5.896	3.73e-09 ***
glycemie	-0.009318	0.002075	-4.491	7.09e-06 ***
hta[T.0]	1.525914	0.202102	7.550	4.35e-14 ***
hypgl[T.1]	-1.583185	0.203851	-7.766	8.08e-15 ***
hypgly[T.1]	-1.414361	0.231974	-6.097	1.08e-09 ***
hypten[T.1]	0.377186	0.251536	1.500	0.13374
imc	0.042959	0.015648	2.745	0.00605 **
nouvhta[T.1]	-0.563805	0.352815	-1.598	0.11004

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.495257	0.567873	0.872	0.3831
antediab[T.0]	1.404397	0.145390	9.660	< 2e-16 ***
dyslipid[T.0]	1.590301	0.268872	5.915	3.32e-09 ***
glycemie	-0.009452	0.002081	-4.543	5.55e-06 ***
hta[T.0]	1.344559	0.159627	8.423	< 2e-16 ***
hypgl[T.1]	-1.651784	0.194824	-8.478	< 2e-16 ***
hypgly[T.1]	-1.414429	0.231570	-6.108	1.01e-09 ***
imc	0.035632	0.014860	2.398	0.0165 *

Le model

p (y=1/x)= 0.495257 +1.404397 antediab[T.0]-0.009452glycemie + 1.344559 hta[T.0]- 1.651784hypgl[T.1]+ -1.414429hypgly[T.1]+ 0.035632 imc Nous calculons l' odd ratio :

antediab[T.0]	dyslipid[T.0]	glycemie	hta[T.0]	hypgl[T.1]	hypgly[T.1]	imc
4.07307	4.905225	0.9905925	3.836494	0.1917076	0.2430644	1.036274

- Pour la variable antediab[T.0] : Le risque d'avoir un diabétique chez un individu qui consomme les dattes est multiplié par4.07307.
- Pour la variable dyslipid[T.0] : Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pêches est multiplié par 4.905225.
- Pour la variable glycemie : Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pommes est multiplié par 0.9905925. hta[T.0] - Pour la variable hta[T.0] : Le risque d'avoir un diabétique chez un individu qui consomme les dattes est multiplié par 3.836494.
- Pour la variable hypgl[T.1] : Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pêches est multiplié par 0.1917076.
Pour la variable hypgly[T.1] : Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pommes est multiplié par 0.2430644. - Pour la variable imc :

Le risque d'avoir un diabétique chez un individu qui ne consomme pas les pommes est multiplié par 1.036274

application arbre de décision

Application de la méthode RPART :
1) root 3372 264 0 (0.07829181 0.92170819)
2) glycemie>=117.5 356 169 1 (0.52528090 0.47471910)
4) glycemie>=139.5 233 76 1 (0.67381974 0.32618026)
8) hta=1 95 17 1 (0.82105263 0.17894737) *
9) hta=0 138 59 1 (0.57246377 0.42753623)
18) imc< 30.26452 105 35 1 (0.66666667 0.33333333)

74) antehta=0 34 8 1 (0.76470588 0.23529412) *
75) antehta=1 29 10 0 (0.34482759 0.65517241)
150) triglyce>=145 11 4 1 (0.63636364 0.36363636) *
151) triglyce< 145 10 1 0 (0.10000000 0.90000000) *
19) imc>=30.26452 31 8 0 (0.25806452 0.74193548) *
5) glycemie< 139.5 123 30 0 (0.24390244 0.75609756) *
3) glycemie< 117.5 2991 76 0 (0.02540956 0.97459044) *

Erreur de noeud racine :264/3372 = 0.078292.

	CP	nsplit	rel error	xerror	xstd
1	0.155303	0	1.00000	1.00000	0.059087
2	0.030303	2	0.68939	0.72727	0.050970
3	0.017045	4	0.62879	0.70455	0.050215
4	0.010000	7	0.57576	0.69697	0.04996

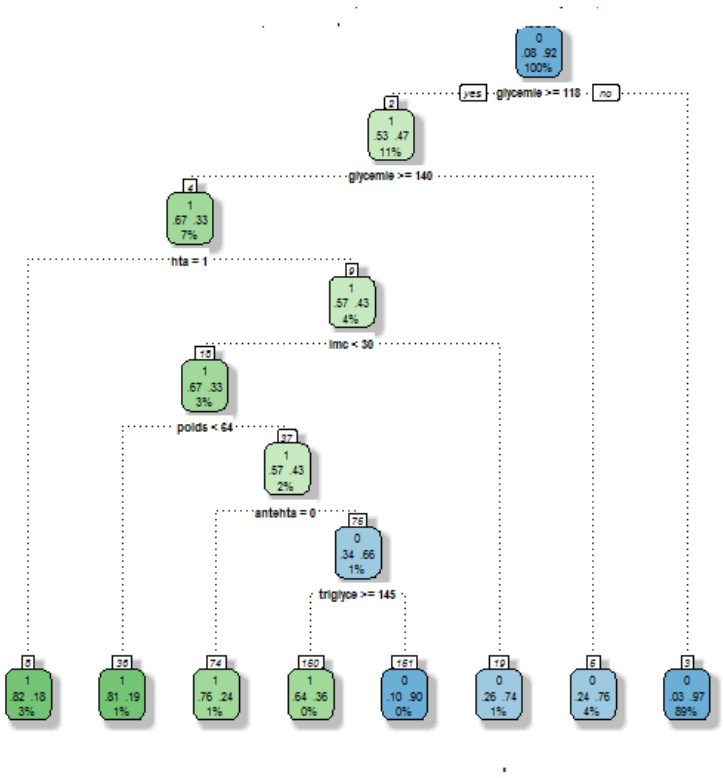

FIGURE VI.1 – arbre de décision data datam /diabète

Discussion :

Ces résultats obtenus sur l'étude de 4818 individus recensés en Algérie ont montré la glycemie , la tension artérielle et imc sont fortement corrélés avec le risque du diabète.

	CP	nsplit	rel error	xerror	xstd
1	0.016	0	1.00	1.00	0.13628
2	0.012	7	0.80	1.10	0.14238
3	0.010	16	0.68	1.16	0.14587

réseaux de neurones

les poids $w_{i.j}$

b->h1	i1->h1	i2->h1	i3->h1	i4->h1	i5->h1	i6->h1	i7->h1	i8->h1
-0.66	0.23	0.29	-0.31	-0.68	-0.36	0.27	0.23	-0.31

La variable cible (diabète) étant binaire, de fait, R n'en décrit qu'une seule. Nous obtenons le réseau suivant :
Appliquée sur l'échantillon test (2409 observations) et le réseau construit dans la section précédente :
Nous obtenons un taux d'erreur de 0.8214317.

Conclusion :

Les résultats sont conformes aux résultats émis par les spécialistes en diabétologie, basés sur d'autres méthodes statistiques :

- Pour la variable hta(tension inertielle est un facteur prédictif du diabète.
-
- la glucemie et imc (Indice de masse corporelle)sont des mesures médical susceptibles d'être touchés par le du diabète.
- antediab(Antécédents familiaux de diabète) sont prédisposés au diabète.

CHAPITRE VII

ETUDE COMPARATIVE DE MODÈLES DE PRÉVISION SUR DONNÉES ENQUÊTES :

I. La courbe ROC

1. Introduction

La courbe ROC (Fawcett) [32] offre à la fois une vision graphique et une mesure pertinentes de la performance d ?un classifier. Elle possède de nombreux avantages par rapport aux mesures de rappel et précision par classe :

La performance est synthétisée par une unique mesure qui ne dépend pas des proportions de classe. Cet avantage se transforme néanmoins en inconvénient lorsqu'il s'agit de revenir rapidement aux mesures de rappel et de précision par classe ou d'estimer le comportement du classifier selon les classes.

Les mesures de rappel et précision sont en effet utiles car elles caractérisent précisément

Le comportement du classifier sur chacune des classes. En particulier, lorsque les classes sont déséquilibrées, ces mesures fournissent des indications de performance plus représentatives et concrètes qu'un unique score pour le classifier.

Notre travail vise à améliorer l'estimation des performances d'un classifier dont on connaît la courbe ROC : il s'agit donc d'une méthode graphique d'estimation, qui d'un coup d' ?il permet de lire sur la courbe les rappel, précision et Fscore par classe. Bien évidemment, un calcul adapté, étant donnés les points de la courbe, permettrait de préciser tout score. Nous souhaitons plutôt introduire un mode nouveau de lecture de la courbe ROC qui permette d'appréhender plus profondément le comportement du classifier. Par exemple, s'il est aisé de mesurer graphiquement le rappel en chaque point de la courbe (c'est son ordonnée), la valeur de la précision ne peut être obtenue que par calcul. Une méthode graphique pour la déterminer rapidement serait la bienvenue.

II. Interprétation géométrique de la courbe ROC

La courbe ROC est une caractéristique de la performance indépendante de la classe, cependant elle représente le point de vue de la classe positive, généralement la classe majoritaire.

Des mesures concernant la classe minoritaire sont donc difficiles à obtenir graphiquement.

Nous présentons donc une méthode simple d'interprétation graphique de la courbe ROC,tant d'un point de vue de la classe positive que négative. Des considérations graphiques sur le point d'intersection de la courbe avec la diagonale descendante fournissent simplement des valeurs fiables du rappel et de la précision pour chaque classe.

III. Préliminaires

Dans un contexte d'apprentissage supervisé, qui étudie des exemples d'apprentissage (u_i) pourvus d ?une étiquette indiquant sa classe, nous focalisons notre attention sur les problèmes de classification binaire ou conceptuelle [Cornuéjols A. (2010)] : il n'y a que deux classes,notées P et$N$, contenant respectivement P et N exemples. Le résultat de la classification est noté dans une table de contingence :

Décision Vérité	P	N	total
P	TP	FP	P
N	FN	TN	N
total	P	N	P+ N

Le résultat d'un classifier utilise : TP (resp. TN) : le nombre de vrais positifs (resp. négatifs) FP (resp. FN) : le nombre de faux positifs (resp. négatifs).
Les mesures traditionnelles de la performance en classification sont propres à chaque classe :
Fscore : le Fscore est une moyenne harmonique de paramètre β :

. Dans la suite on considérera le cas $\beta = 1$

Pour obtenir des rappel, précision et Fscore indépendants des classes, on effectue les moyennes des valeurs par classe. Une autre mesure traditionnelle est le score de classification, qui indique la proportion d'exemples bien classés (dans les deux classes).

IV. Aire sous la courbe ROC

L('utilisation des mesures précédemment décrites est contestée [Provost et al. (1998)], notamment car elles fournissent une vue trop particulière de la performance du clas-

sifier et parce qu'elles sont trop sensibles à la disproportion de classe. On leur préfère souvent l'aire sous la courbe ROC, qui utilise des mesures normalisées par les populations des classes : les taux de vrais ou de faux positifs.

Considérant que le classifier rend sa décision sous la forme d'une probabilité d'appartenance à la classe P, la décision finale est modulaire par un seuil. En faisant varier ce seuil, on Calcule pour chaque classification le taux de faux positifs TP/Net celui de vrais positifsTP/P , qui définissent la courbe dite ROC. L'aire sous cette courbe indique la probabilité que le classifier fournisse un score d ?appartenance à P supérieur à celui d'appartenir à N.

V. Méthode graphique d'interprétation de la courbe ROC

Notre travail vise à mieux cerner les différences de classification selon les classes grâce une lecture graphique de la courbe ROC. Nous considérons pour cela disposé de la courbe ROC fournie par un classifier, et nous déduisons des informations de cette courbe pour des constructions graphiques.
Réponse binaire - Courbe ROC On se place dans le cas d'une réponse binaire (+/-). On obtient un résultat de prédiction de la forme + FN TP P

O	(-)	(+)	Total
(-)	TN	FP	N
(+)	FN	TP	P

Avec

TP (true positives) : les prédits positifs qui le sont vraiment.

- FP (false positives) : les prédits positifs qui sont en fait négatifs.
- TN (true negatives) : les prédits négatifs qui le sont vraiment.
- FN (false negatives) : les prédits négatifs qui sont en fait positifs.
- P (positives) : tous les positifs quel que soit l'état de leur prédiction $P = TP + FN$.
- N (negatives) : tous les négatifs quel que soit l'état de leur prédiction $N = TN + FP$.

On définit alors la spécificité et la sensibilité :

- la sensibilité est :$TP/(TP + FN) = TP/P$.
- la spécificité est :$TN/(TN + FP) = TN/N$.

Principe de la courbe ROC : Si le test donne un résultat numérique avec un seuil t tel que la prédiction est positive si$x > t$, et la prédiction est négative si $x < t$, alors au fur et à mesure que t augmente :

– la spécificité augment.
– mais la sensibilité diminue.

La courbe ROC représente l'évolution de la sensibilité (taux de vrais positifs) en fonction de
– spécificité (taux de faux positifs) quand on fait varier le seuil t.
– C'est une courbe croissante entre le point (0,0) et le point (1, 1) et en principe au-dessus de

la première bissectrice.
– Une prédiction random donnerait la première bissectrice.
– Meilleure est la prédiction, plus la courbe est au-dessus de la première bissectrice.
– Une prédiction idéale est l'horizontale$y = 1$ sur]0,1] et le point (0,0).
– L'aire sous la courbe ROC (AUC, Area Under the Curve) donne un indicateur de la qualité de la prédiction (1 pour une prédiction idéale, 0.5 pour une prédiction random).

VI. Conclusion

Nous avons proposé une méthode simple d'interprétation des performances d'un classifier étant donnée sa courbe ROC. L'ordonnée des points d'intersection avec des diagonales descendantes fait le lien avec le rappel, la précision et le Fscore par classes. Cette construction permet d'appréhender graphiquement les différences de performance selon les classes et d'approfondir la lecture de la courbe ROC.

VII. Application

pour (variables dictomies)

Application :on applique le régression logistique et l'arbre de décision ,réseau neurone pour compare les trois modelé
la zone sous la courbe ROC pour le modèle rpart sur datam [validation de] est de0.7911
La zone sous la courbe ROC pour le modèle glm sur datam [validation de] est de 0.8984
La zone sous la courbe ROC pour le modèle nnet sur datam [validation de] est de 0.5000

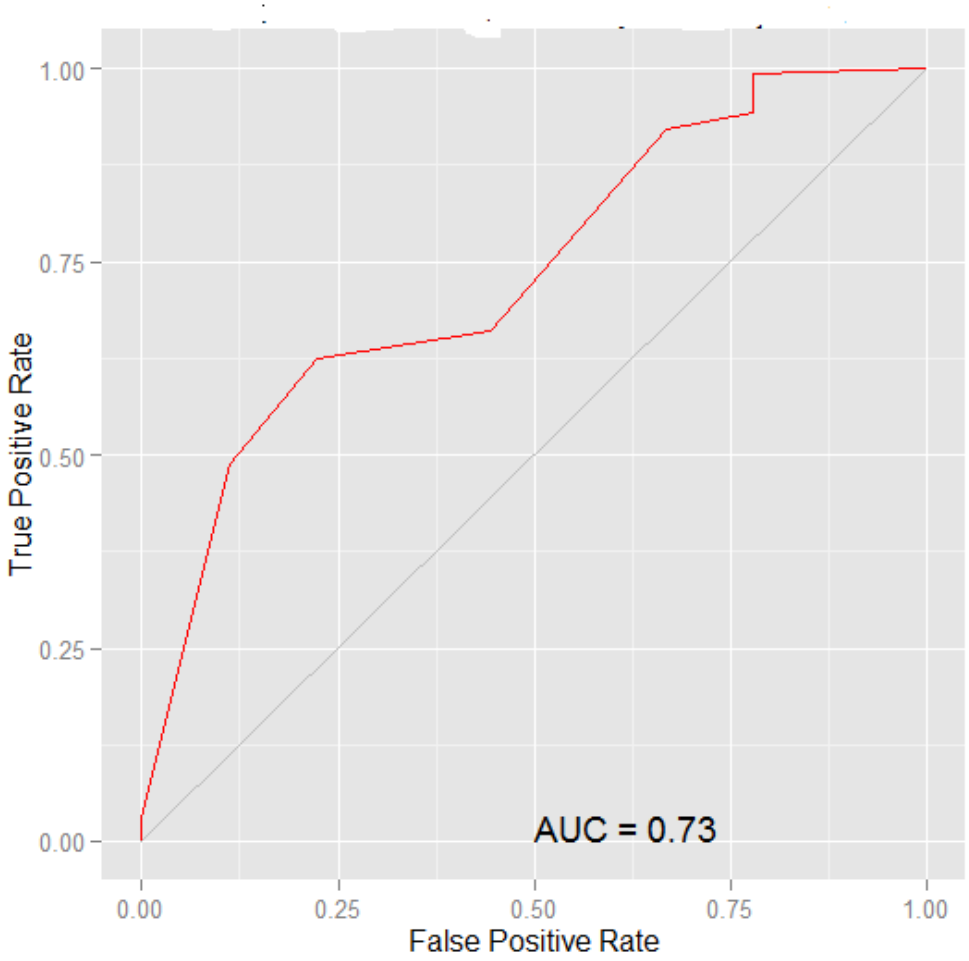

FIGURE VII.1 – ROC Curve Réseaux de neurones data1[validation]diabète

nous souhaitons évaluer les performances de nos trois modèles prédictifs sur notre échantillon test Nous passons à l'onglet « Evaluâtes ». Dans un premier temps, nous demandons la matrice de Confusion. Enfin, les trois modèles que nous avons élaborés dans la section précédente sont automatiquement sélectionnés. Nous pouvons évaluer qu'une partie d'entre eux, nous ne pouvons pas en revanche tester un modèle qui n'aurait pas été construit dans l'onglet model. Pour évaluer leur capacité à scorer c.àd. à attribuer un score plus élevé aux positifs par rapport au négatifs, nous utilisons la courbe ROC.Si l'on se réfère aux courbes et à l'indicateur AUC (Area under curve),

Nous constatons que la Régression logistique et l' arbre de décision sont proches, le réseau neurone est un peu en retrait.

pour deuxième modelé polytomies

Application :on applique le régression logistique et l'arbre de décision ,réseau neurone pour compare les trois modelé

la zone sous la courbe ROC pour le modèle rpart sur data2 [validation de] est de0.5000.

La zone sous la courbe ROC pour le modèle glm sur data2 [validation de]

est de 0.6189.
La zone sous la courbe ROC pour le modèle nnet sur data2 [validation de] est de 0.6183.

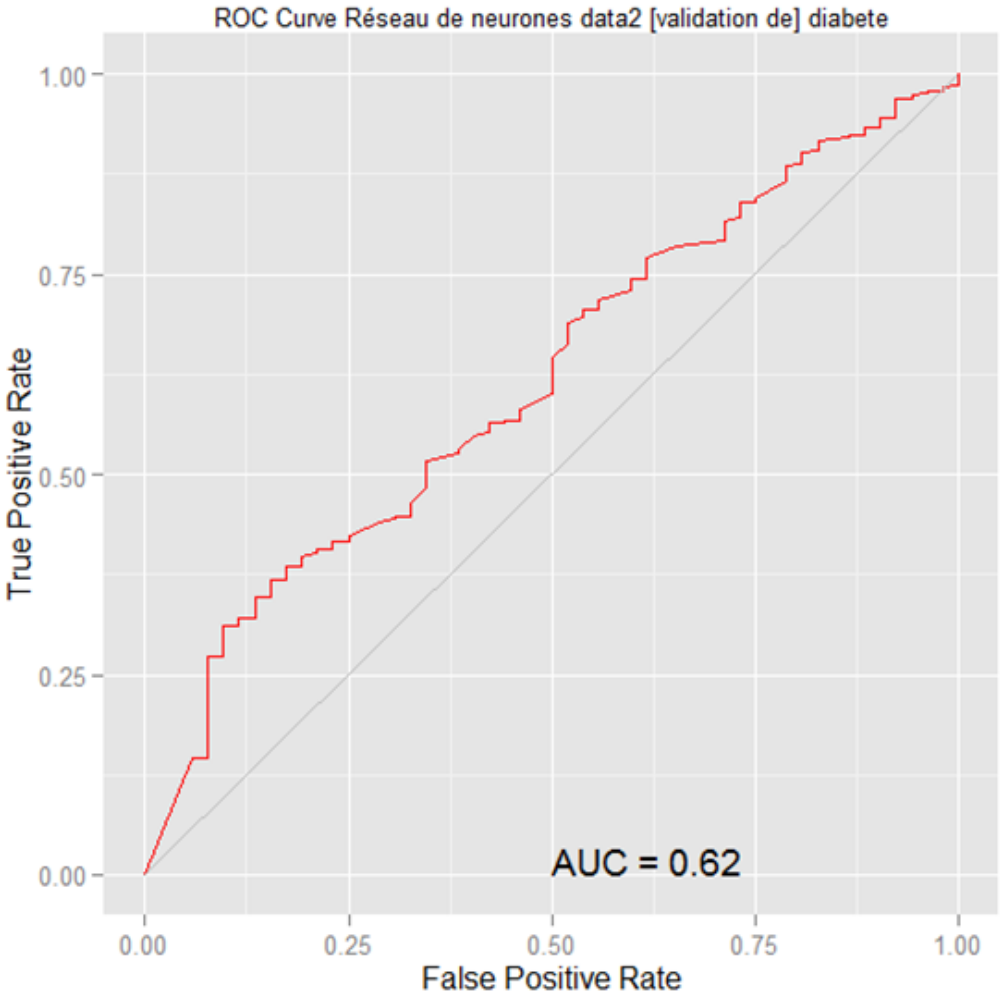

FIGURE VII.2 – ROC Curve Réseaux de neurones data2 [validation]diabète

Nous constatons que la Régression logistique et le réseau neurone sont proches, l' arbre de décision est un peu en retrait.

VIII. Prévision

Comme tout modèle par logiciel R, un arbre obtenu par rpart (avec ou sans simplification) peut réaliser des prévisions sur de nouvelles données, en s'appuyant sur la fonction predict. Par défaut, la fonction estime les probabilités d'appartenance aux classes pour chaque observation (simplement par le ratio dans la feuille correspondante).

Nous obtenons par exemple on donne les estimations suivantes sur les dix premiers individus aléatoire $N = 1, ...10$.

N	diabétique (1)	non diabétique (0)
2337	0.06339468	0.9366053
2449	0.06339468	0.9366053
3193	0.06339468	0.9366053
1391	0.06339468	0.9366053
4340	0.80339468	0.066053
1505	0.06339468	0.9366053
4231	0.06339468	0.9366053
3926	0.06339468	0.9366053
3827	0.7549468	0.0686053
4480	0.06339468	0.9366053

on remarque que les individus (2337,2449,3193,1391,1505,4231,3926,4480) ne risque pas d'être diabétique en futur , par contre ,pour les individus (4340,3827) sont prédites d'être diabétique.

CONCLUSION ET PERSPECTIVES

L'objectif principal de ce mémoire est d'expliquer les facteurs de risque du diabète en Algérie et de déterminer l'impact du diabète sur les maladies chroniques et ce, sur la base des données de l'enquête nationale de santé « TAHINA » intitulé « Étude du diabète en Algérie ».

Ainsi nous avons exploité l'ensemble des données et avons déroulés plusieurs applications.

Parmi elle l'application de la régression logistique dans le chapitre II qui a montré que :

1. les individus dont le régime alimentaire est basé sur les légumes et les fruits ont moins de risque d'avoir un diabète par rapport à ceux dont le régime alimentaire est basé sur les volailles à l'exception de la dinde ;

2. Le diabète est moins constaté chez les fumeurs et les individus dont le régime alimentaire est basé sur les légumes ;

3. L'âge est un facteur de risque de diabète.

Nous avons présenté aussi dans ce mémoire des systèmes d'apprentissage basés sur les méthodes de segmentation : arbre de décision et réseaux de neurones ; A ce titre, il nous semble opportun de rappeler que pour l'analyse de données, les méthodes statistiques restent très importantes et sont largement utilisées dans plusieurs domaines.

L'apprentissage par les méthodes de segmentation fait partie de la classe des méthodes non-paramétriques et sont issues de l'intelligence artificielle.

A partir des données fournies, nous avons appliqué les méthodes de segmentation par le biais de l'utilisation du logiciel arbre de décision basé sur la méthode CART.

Ceci a permis un tracé rapide du meilleur chemin à proposer aux médecins praticiens soucieux d'une prise en charge effective du diabétique. Les résultats obtenus par le logiciel sont faciles à interpréter. Le diagramme de l'arbre permet d'identifier rapidement les plus importants prédicteurs.

Ainsi, le praticien disposant de cet outil d'aide, pourra l'utiliser comme moyen d'expertise de son diagnostic et de la future prise en charge thérapeutique.

Enfin les résultats obtenus au cours de cette étude méritent d'être poursuivis et appliqués pour d'autres domaines.

Concernant les réseaux, il existe plusieurs types et pour chacun quelques paramètres ont été modifiés afin d'améliorer la précision. Les résultats obtenus montrent que les réseaux multicouches ayant une seule couche cachée et quelques neurones à l'intérieure de chaque couche est un approximateur universelle parcimonieux pour toute fonction suffisamment régulière dans son domaine de définition.

Le problème médical à l'aide des réseaux de neurones a été abordé. Ainsi, le changement effectué sur le nombre de neurone d'entrée et le nombre de neurones dans la couche cachée, a montré que le nombre d'itération l'erreur n'a pas diminué.

Les travaux réalisés dans le présent mémoire ont permis de trouver qu'un réseau à 33 entrées et 02 neurones pour les variables dichotomies et 8 entrées et 13 neurones pour et variables polytomies de la couche cachée suffisent pour la modélisation du diagnostic d'un malade atteint du diabète.

Cette thèse n'a pas la prétention de résoudre le problème de l'application de la segmentation, mais traite de l'application des réseaux de neurones aux données binaires, ce qui fait son originalité. Elle propose un outil d'aide à la décision, voire de modélisation des données binaires.

Les perspectives à envisager concernent la nature des données. Ainsi l'application pourrait être élargie aux autres composantes du diagnostic d'un malade. La diversité des données pourrait contribuer au raffinement de la modélisation.

Il est à signaler que diverses méthodes peuvent être associées aux réseaux de neurones pour améliorer la précision des modèles et nous pouvons citer : la logique floue, les chaînes de Markov cachées et les algorithmes génétiques.

Enfin, nous envisageons d'étendre cette étude au cas des arbres de décision non binaire. Cela va nous permettre en outre l'introduction d'autres coefficients d'association entre variables et faire une étude comparative

avec des données médicales et dans un autre cadre de recherche, nous pensons et souhaiterions étendre notre travail à un arbre neuronal.

ANNEXE

analyse composante principe

Variables	painjc	couscou	patejc	rizjc	cerealj	pomterr	legusec	fruitj	legume	laitjc	poisson
painjc	1	0,012	0,028	0,100	-0,062	0,196	0,009	0,095	0,165	0,224	0,094
couscou1jc	0,012	1	0,308	0,198	0,223	0,091	0,232	0,045	0,098	-0,046	-0,017
patejc	0,028	0,308	1	0,370	0,224	0,052	0,230	0,018	0,035	0,024	0,020
rizjc	0,100	0,198	0,370	1	0,234	0,094	0,182	0,050	0,114	0,036	0,081
cerealjc	-0,062	0,223	0,224	0,234	1	0,008	0,230	0,118	0,065	0,010	0,063
pomterrjc	0,196	0,091	0,052	0,094	0,008	1	0,038	0,122	0,326	0,141	0,006
legusecjc	0,009	0,232	0,230	0,182	0,230	0,038	1	0,103	0,106	-0,026	0,149
fruitjc	0,095	0,045	0,018	0,050	0,118	0,122	0,103	1	0,374	0,191	0,103
legumejc	0,165	0,098	0,035	0,114	0,065	0,326	0,106	0,374	1	0,154	0,092
laitjc	0,224	-0,046	0,024	0,036	0,010	0,141	-0,026	0,191	0,154	1	0,012
poissonjc	0,094	-0,017	0,020	0,081	0,063	0,006	0,149	0,103	0,092	0,012	1
viandejc	0,004	0,157	0,161	0,176	0,170	0,020	0,143	0,235	0,137	0,058	0,077
volailljc	0,110	0,088	0,072	0,141	0,131	0,099	0,069	0,200	0,142	0,087	0,150
oeufjc	0,040	0,013	0,039	0,108	0,066	0,178	0,050	0,161	0,174	0,165	0,061
cacahuejc	0,019	0,060	0,112	0,102	0,151	-0,014	0,098	0,076	0,020	0,081	0,049
dessertjc	0,166	-0,009	0,046	0,094	0,066	0,169	0,039	0,244	0,204	0,165	0,102
huilolijc	0,069	-0,077	-0,042	-0,093	-0,078	-0,013	-0,018	0,090	0,015	0,179	0,036
huilejc	0,239	0,098	0,095	0,109	0,035	0,237	0,069	0,130	0,225	0,193	0,038
beurrejc	0,155	0,236	0,162	0,187	0,137	0,088	0,133	0,148	0,138	0,128	0,063
age	-0,001	0,049	-0,037	-0,041	0,014	-0,020	0,005	0,013	-0,019	-0,040	-0,019
feculenj	0,772	0,390	0,373	0,387	0,234	0,578	0,175	0,151	0,297	0,209	0,086

viandejc	volaill	oeufjc	cacahue	dessert	huilolij	huilejc	beurre	age	feculen
0,004	0,110	0,040	0,019	0,166	0,069	0,239	0,155	-0,001	0,772
0,157	0,088	0,013	0,060	-0,009	-0,077	0,098	0,236	-0,049	0,390
0,161	0,072	0,039	0,112	0,046	-0,042	0,095	0,162	-0,037	0,373
0,176	0,141	0,108	0,102	0,094	-0,093	0,109	0,187	-0,041	0,387
0,170	0,131	0,066	0,151	0,066	-0,078	0,035	0,137	-0,014	0,234
0,020	0,099	0,178	-0,014	0,169	-0,013	0,237	0,088	-0,020	0,578
0,143	0,069	0,050	0,098	0,039	-0,018	0,069	0,133	-0,005	0,175
0,235	0,200	0,161	0,076	0,244	0,090	0,130	0,148	-0,013	0,151
0,137	0,142	0,174	0,020	0,204	0,015	0,225	0,138	-0,019	0,297
0,058	0,087	0,165	0,081	0,165	0,179	0,193	0,128	-0,040	0,209
0,077	0,150	0,061	0,049	0,102	0,036	0,038	0,063	-0,019	0,086
1	0,241	0,110	0,131	0,173	-0,002	0,097	0,156	-0,006	0,125
0,241	1	0,157	0,062	0,155	-0,019	0,021	0,152	-0,010	0,186
0,110	0,157	1	0,023	0,188	0,051	0,107	0,221	-0,068	0,133
0,131	0,062	0,023	1	0,050	0,024	0,058	0,109	-0,064	0,078
0,173	0,155	0,188	0,050	1	0,005	0,211	0,151	-0,093	0,209

FIGURE VII.3 – les individus

régression logistique

Groupe2 : ce groupe contient 28 variables qualitatives «consommation de fruits» :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.788e+00	3.599e-01	16.088	< 2e-16 ***
abricot[T.0]	-9.980e-02	1.228e-01	-0.813	0.41630
ananas[T.0]	-1.388e+01	4.610e+02	-0.030	0.97598
banane[T.0]	-4.769e-01	3.351e-01	-1.423	0.15470
cerise[T.0]	-1.384e+01	4.202e+02	-0.033	0.97373
citron[T.0]	1.087e-01	4.459e-01	0.244	0.80742
coing[T.0]	-1.401e+01	8.573e+02	-0.016	0.98696
datte[T.0]	-1.430e+00	4.679e-01	-3.056	0.00224 **
figuebar[T.0]	-1.402e+01	5.468e+02	-0.026	0.97954
figuefr[T.0]	-7.038e-02	2.433e-01	-0.289	0.77233
fraise[T.0]	-4.908e-01	7.346e-01	-0.668	0.50405
grenade[T.0]	-1.397e+01	1.677e+03	-0.008	0.99335
jujube[T.0]	-1.437e+01	2.400e+03	-0.006	0.99522
kiwi[T.0]	-1.442e+01	2.400e+03	-0.006	0.99521
mandarin[T.0]	-1.250e+01	2.400e+03	-0.005	0.99584
mangue[T.0]	6.651e-01	1.088e+00	0.611	0.54097
margarin[T.0]	4.304e-03	1.758e-01	0.024	0.98047
melon[T.0]	-4.720e-01	3.340e-01	-1.413	0.15758
nectarin[T.0]	9.226e-01	7.906e-01	1.167	0.24323
nefle[T.0]	2.636e-01	2.598e-01	1.015	0.31034
orange[T.0]	-2.743e-01	1.037e+00	-0.264	0.79150
pamplemo[T.0]	1.130e+00	1.103e+00	1.024	0.30589
pasteque[T.0]	4.679e-02	1.841e-01	0.254	0.79934
peche[T.0]	3.220e-01	1.302e-01	2.474	0.01336 *
poire[T.0]	-6.426e-01	7.324e-01	-0.877	0.38026
pomme[T.0]	7.509e-01	1.725e-01	4.353	1.34e-05 ***
prune[T.0]	-2.101e-01	2.135e-01	-0.984	0.32496
pruneaux[T.0]	-6.796e-02	7.468e-01	-0.091	0.92748
raisin[T.0]	2.668e-01	6.252e-01	0.427	0.66963

Après avoir éliminé les variables ayons un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau si dessus :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.9656	0.4820	6.152	7.64e-10 ***
datte[T.1]	-1.4519	0.4566	-3.180	0.00148 **
peche[T.0]	0.3019	0.1283	2.353	0.01863 *
pomme[T.0]	0.7258	0.1682	4.314	1.60e-05 ***

Groupe 3 : groupe des légumes contient 35 variables qualitatives

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	25.82044	2425.09408	0.011	0.991505
artichau[T.0]	0.22266	0.76687	0.290	0.771546
asperge[T.0]	0.89123	0.69813	1.277	0.201745
aubergin[T.0]	0.24416	0.24769	0.986	0.324246
avocat[T.1]	2.19041	0.78180	2.802	0.005083 **
betterav[T.1]	-0.69074	0.29356	-2.353	0.018624 *
blette[T.0]	0.05271	1.06516	0.049	0.960535
cardon[T.0]	0.33555	0.44090	0.761	0.446623
carotte[T.0]	-0.15612	0.14321	-1.090	0.275642
caroube[T.0]	0.45798	2527.24043	0.000	0.999855
celeribr[T.0]	0.48958	0.31013	1.579	0.114422
choufleu[T.0]	0.41729	0.43563	0.958	0.338105
chouvert[T.0]	-14.58484	351.09028	-0.042	0.966864
concombr[T.0]	0.12063	0.12846	0.939	0.347669
courge[T.0]	-0.07474	0.27784	-0.269	0.787920
courgett0[T.0]	0.09234	0.12241	0.754	0.450674
epinards[T.0]	-0.59318	0.53916	-1.100	0.271249
fenouil[T.0]	1.09607	0.83855	1.307	0.191178
feve[T.0]	-0.16900	0.44441	-0.380	0.703738
frites[T.1]	-0.26070	0.12653	-2.060	0.039367 *
gombo[T.0]	0.78280	1.17740	0.665	0.506141
grcourge[T.0]	-14.13924	793.17666	-0.018	0.985778
haricbla[T.0]	0.37623	0.30547	1.232	0.218090
haricver[T.0]	0.03956	0.15562	0.254	0.799316
navet[T.0]	-0.05142	0.25370	-0.203	0.839374
oignon[T.0]	-0.19927	0.12945	-1.539	0.123718
olive[T.0]	0.05517	0.15656	0.352	0.724551
patatedo[T.0]	-0.41102	0.52496	-0.783	0.433651
petitpoi[T.0]	0.23062	0.24369	0.946	0.343974
piment[T.0]	0.18698	0.13261	1.410	0.158549
poireau[T.0]	0.02366	0.33201	0.071	0.943183
poivron[T.0]	-0.18453	0.13722	-1.345	0.178682
radis[T.0]	-0.33128	0.43820	-0.756	0.449647
saladver[T.0]	0.40916	0.12089	3.384	0.000713 ***
hline tomateco[T.0]	0.03960	0.12457	0.318	0.750545
tomatefr[T.0]	0.03742	0.13440	0.278	0.780699
variante[T.0]	-0.15073	0.26933	-0.560	0.575713

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.0571	0.8188	1.291	0.1967
avocat[T.1]	1.9861	0.7716	2.574	0.0101 *
betterav[T.1]	-0.6395	0.2724	-2.347	0.0189 *
frites[T.1]	-0.2589	0.1237	-2.094	0.0363 *
saladver[T.0]	0.4257	0.1089	3.908	9.32e-05 ***

Groupe 5 : ce groupe contient 18 variables qualitatives des différents types des protéines (viandes et volailles et ....)

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	4.277e+01	1.589e+03	0.027	0.9785
abats[T.0]	-4.299e-02	4.009e-01	-0.107	0.9146
boeuf[T.0]	2.726e-01	1.776e-01	1.535	0.1248
brebis[T.0]	7.495e-02	6.095e-01	0.123	0.9021
cachir[T.0]	-4.119e-01	5.228e-01	-0.788	0.4308
chameau[T.0]	-1.799e+00	1.012e+00	-1.778	0.0755 .
crustace[T.0]	-1.212e+01	4.773e+02	-0.025	0.9797
dinde[T.0]	6.112e-01	3.465e-01	1.764	0.0777 .
kaddid[T.0]	-3.579e-01	7.330e-01	-0.488	0.6253
lapin[T.0]	8.794e-01	7.886e-01	1.115	0.2648
mouton[T.1]	3.940e-01	1.626e-01	2.423	0.0154 *
oeufs	-1.751e-03	1.012e-01	-0.017	0.9862
poisscon[T.0]	-1.723e+00	1.011e+00	-1.704	0.0883 .
poissech[T.0]	-1.249e+01	4.213e+02	-0.030	0.9764
poissfra[T.0]	2.923e-01	1.860e-01	1.572	0.1160
poulet[T.0]	2.501e-01	1.434e-01	1.744	0.0812 .
cheval[T.0]	-1.302e+01	1.455e+03	-0.009	0.9929
chevre[T.0]	-3.454e-03	4.709e-01	-0.007	0.9941
merguez[T.0]	-1.029e+00	1.020e+00	-1.008	0.3135

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio

Coefficients	Estimate Std. Error	z value	Pr(>|z|)	
(Intercept)	5.0844	1.4573	3.489	0.000485 ***
chameau[T.0]	-1.8466	1.0070	-1.834	0.066694 .
dinde[T.0]	0.5801	0.3446	1.684	0.092251 .
mouton[T.1]	0.3421	0.1605	2.131	0.033053 *
poulet[T.0]	0.2548	0.1418	1.797	0.072315 .
poisscon[T.0]	-1.9005	1.0091	-1.883	0.059648 .

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.6155	1.4242	3.943	8.05e-05 ***
chameau[T.0]	-1.8600	1.0070	-1.847	0.0647 .
poisscon[T.0]	-1.8449	1.0076	-1.831	0.0671 .
poulet[T.0]	0.2470	0.1417	1.744	0.0812 .
mouton[T.1]	0.3442	0.1604	2.146	0.0319 *

Après avoir éliminé les variables ayons un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.8502	1.4199	4.120	3.79e-05 ***
chameau[T.0]	-1.8895	1.0068	-1.877	0.0605 .
poisscon[T.0]	-1.8303	1.0074	-1.817	0.0692 .
mouton[T.1]	0.3305	0.1601	2.064	0.0390 *

Groupe 6 : ce groupe contient 23 variables qualitatives des légumes et fruits secs

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	7.342e+01	1.583e+03	0.046	0.9630
abricsec[T.0]	-7.863e-01	1.035e+00	-0.760	0.4475
amande[T.0]	-1.386e+01	4.539e+02	-0.031	0.9756
cacahuet[T.0]	-2.385e-03	2.531e-01	-0.009	0.9925
figuesec[T.0]	-1.363e+01	6.786e+02	-0.020	0.9840
mermez[T.0]	-4.234e-01	4.270e-01	-0.992	0.3214
noisette[T.0]	-1.427e+01	8.591e+02	-0.017	0.9867
noix[T.0]	7.750e-01	1.094e+00	0.708	0.4787
pate[T.0]	-1.423e+01	9.660e+02	-0.015	0.9882
pistache[T.0]	-1.384e+01	4.103e+02	-0.034	0.9731
pizza[T.0]	-3.271e-01	3.988e-01	-0.820	0.4122
raisisec[T.0]	3.386e-01	5.467e-01	0.619	0.5357
viennois[T.0]	5.545e-01	2.542e-01	2.182	0.0291 *
riz[T.0]	3.325e-01	1.707e-01	1.948	0.0514 .
poiscass[T.0]	-1.752e+00	1.028e+00	-1.705	0.0882 .
poischic[T.0]	1.292e-01	1.612e-01	0.801	0.4229
moutarde[T.0]	-1.150e-02	6.222e-01	-0.018	0.9852
margarin[T.0]	-2.694e-02	1.768e-01	-0.152	0.8789
lentille[T.0]	3.077e-02	2.370e-01	0.130	0.8967
harissa[T.0]	-3.063e-01	2.668e-01	-1.148	0.2510
haricsec[T.0]	-2.186e-01	3.734e-01	-0.585	0.5583
frick[T.0]	1.267e-02	2.311e-01	0.055	0.9563
dchicha[T.0]	5.948e-01 2.442e-01 2.436	0.0149 *		

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.0376	1.0535	2.883	0.00393 **
viennois[T.0]	0.4907	0.2513	1.953	0.05086 .
riz[T.0]	0.2760	0.1680	1.642	0.10050
poiscass[T.0]	-1.9432	1.0179	-1.909	0.05626 .
poischic[T.0]	0.1166	0.1584	0.736	0.46164
dchicha[T.0]	0.5579	0.2367	2.357	0.01843 *

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.1603	1.0519	3.004	0.00266 **
viennois[T.0]	0.4901	0.2511	1.952	0.05097 .
poiscass[T.0]	-1.8378	1.0143	-1.812	0.07000 .
poischic[T.0]	0.1243	0.1582	0.786	0.43204
dchicha[T.0]	0.5718	0.2363	2.420	0.01553 *

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.2328	1.0480	3.085	0.00204 **
viennois[T.0]	0.4961	0.2510	1.977	0.04808 *
poiscass[T.0]	-1.8171	1.0138	-1.792	0.07309 .
dchicha[T.0]	0.5821	0.2359	2.468	0.01359 *

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.5168	0.3342	4.538	5.67e-06 ***
viennois[T.0]	0.4956	0.2508	1.976	0.0482 *
dchicha[T.0]	0.4842	0.2338	2.071	0.0384 *

Groupe7 : le groupe contient des variables qualitatives 17 concernant les différents types de gâteaux et les sucres.

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.88026	1.96485	2.993	0.002765 **
barrecho[T.0]	-1.20822	1.02898	-1.174	0.240318
beignet[T.0]	0.55789	0.55440	1.006	0.314267
beurre[T.0]	0.02073	0.18409	0.113	0.910363
biscotte[T.0]	-0.59611	0.61361	-0.971	0.331314
biscuits[T.0]	0.02527	0.25090	0.101	0.919791
cake[T.0]	0.42012	0.30133	1.394	0.163255
confitur[T.0]	-1.06977	0.29502	-3.626	0.000288 ***
crepes[T.0]	0.55236	0.42699	1.294	0.195800
gatotrad[T.0]	-0.28967	0.28108	-1.031	0.302754
mayonnai[T.0]	0.07639	0.40893	0.187	0.851809
msemmen[T.0]	0.29279	0.39665	0.738	0.460410
patisser[T.0]	-1.15426	0.52000	-2.220	0.026437 *
smen[T.0]	-0.03528	0.30962	-0.114	0.909269
sucre[T.0]	-1.37357	0.11491	-11.953	< 2e-16 ***
sucrette[T.0]	1.43878	0.21950	6.555	5.57e-11 ***
tarte[T.0]	-0.82472	1.07012	-0.771	0.440896
miel[T.0]	-1.19310	0.73300	-1.628	0.103588

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous et nous calculons le odd ratio :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.6988	0.6114	6.049	1.45e-09 ***
confitur[T.0]	-1.0254	0.2808	-3.651	0.000261 ***
patisser[T.0]	-1.1749	0.5165	-2.275	0.022917 *
sucre[T.0]	-1.3900	0.1139	-12.205	< 2e-16 ***
sucrette[T.0]	1.4063	0.2160	6.510	7.49e-11 ***

Groupe 8 :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.93615	0.22653	17.376	< 2e-16 ***
actif[T.inactif]	-1.08108	0.16319	-6.625	3.48e-11 ***
milieu[T.urbain]	-0.34562	0.12172	-2.840	0.00452 **
nbpieces	-0.09372	0.03061	-3.061	0.00220 **
sexe[T.Hommes]	-0.33717	0.13191	-2.556	0.01059 *
region[T.sud]	0.43589	0.31360	1.390	0.16455
region[T.tell]	-0.02992	0.12016	-0.249	0.80339

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.96540	0.21300	18.617	< 2e-16 ***
actif[T.inactif]	-1.09793	0.16310	-6.731	1.68e-11 ***
milieu[T.urbain]	-0.36966	0.12053	-3.067	0.00216 **
nbpieces	-0.09415	0.03066	-3.071	0.00213 **
sexe[T.Hommes]	-0.33059	0.13202	-2.504	0.01227 *

Le modèle final pour la variable diabète :

Il s'agit de faire la régression logistique avec les variables significatives retenues lors des régressions logistiques précédentes faites par groupe. Résultats

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	4.556e+01	7.474e+03	0.006	0.995137
actif[T.inactif]	-6.048e-01	6.243e-01	-0.969	0.332724
age	-5.440e-02	2.198e-02	-2.475	0.013324 *
avocat[T.0]	2.078e+01	6.523e+03	0.003	0.997459
betterav[T.0]	-1.571e+01	1.178e+03	-0.013	0.989359
boissons	9.650e-05	4.939e-04	0.195	0.845094
exterier[T.0]	9.021e-02	4.166e-01	0.217	0.828574
frites[T.0]	3.346e-01	4.872e-01	0.687	0.492260
jusfruco[T.0]	-5.428e-01	1.133e+00	-0.479	0.631774
legumejc	-2.005e-01	2.022e-01	-0.991	0.321595
limonade[T.0]	-1.470e+00	5.446e-01	-2.699	0.006963 **
milieu[T.urbain]	-3.802e-01	4.073e-01	-0.933	0.350657
mouton[T.0]	1.966e+00	7.116e-01	2.764	0.005718 **
nbpieces	-8.171e-02	1.380e-01	-0.592	0.553752
patisser[T.0]	-1.540e+01	1.775e+03	-0.009	0.993078
peche[T.0]	9.655e-01	5.425e-01	1.780	0.075119 .
poisscon[T.0]	-1.504e+01	2.423e+03	-0.006	0.995047
poissonjc	3.248e-01	5.847e-01	0.555	0.578563
pomme[T.0]	5.352e-01	6.583e-01	0.813	0.416186
puits[T.0]	-1.316e+00	5.755e-01	-2.287	0.022193 *
saladver[T.0]	5.593e-01	4.255e-01	1.315	0.188633
sexe[T.Hommes]	9.061e-02	5.541e-01	0.164	0.870110
sucre[T.0]	-1.526e+00	4.313e-01	-3.539	0.000401 ***
sucrette[T.0]	1.999e+00	7.579e-01	2.637	0.008362 **
viandejc	2.403e-01	4.539e-01	0.530	0.596455
viennois[T.0]	-1.596e+01	1.706e+03	-0.009	0.992532
volailljc	-1.563e-01	4.321e-01	-0.362	0.717531
tabfumsan[T.0]	4.247e-01	6.538e-01	0.650	0.515969

Après avoir éliminé à chaque fois la variable la moins significative nous avons obtenu le tableau suivant :

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	5.467348	0.475956	11.487	< 2e-16 ***
age	-0.065459	0.005769	-11.347	< 2e-16 ***
limonade[T.0]	-0.527523	0.125807	-4.193	2.75e-05 ***
mouton[T.1]	0.325581	0.174692	1.864	0.062358 .
peche[T.0]	0.373315	0.138918	2.687	0.007203 **
puits[T.0]	-0.595898	0.178191	-3.344	0.000825 ***
sucre[T.0]	-1.402585	0.117493	-11.938	< 2e-16 ***
sucrette[T.0]	1.263684	0.226228	5.586	2.33e-08 ***

Régression logistique polytomique

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.820055	0.343123	8.219	<2e-16 ***
wilaya[T.12 :tebes]	0.239013	0.445271	0.537	0.5914
wilaya[T.15 :tizi]	-0.310934	0.425842	-0.730	0.4653
wilaya[T.16 :alger]	-0.698941	0.376246	-1.858	0.0632 .
wilaya[T.18 :jijel]	-0.442569	0.397753	-1.113	0.2658
wilaya[T.21 :skikd]	0.179005	0.433269	0.413	0.6795
wilaya[T.22 :sidi]	-0.806572	0.376631	-2.142	0.0322 *
wilaya[T.23 :annab]	-0.249626	0.395196	-0.632	0.5276
wilaya[T.28 :m'sil]	-0.375423	0.418248	-0.898	0.3694
wilaya[T.29 :masca]	-0.009983	0.420278	-0.024	0.9810
wilaya[T.31 :oran]	-0.722914	0.397283	-1.820	0.0688 .
wilaya[T.33 :illiz]	0.756495	0.612109	1.236	0.2165
wilaya[T.38 :tisse]	0.057158	0.424109	0.135	0.8928
wilaya[T.5 :batna]	-0.664330	0.380721	-1.745	0.0810 .
wilaya[T.7 :biskra]	-0.704509	0.388190	-1.815	0.0695 .
wilaya[T.9 :blida]	-0.051150	0.424555	-0.120	0.9041

regression logistique pour les variables polytomique (wilaya)

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.820055	0.343123	8.219	<2e-16 ***
wilaya[T.12 :tebes]	0.239013	0.445271	0.537	0.5914
wilaya[T.15 :tizi]	-0.310934	0.425842	-0.730	0.4653
wilaya[T.16 :alger]	-0.698941	0.376246	-1.858	0.0632 .
wilaya[T.18 :jijel]	-0.442569	0.397753	-1.113	0.2658
wilaya[T.21 :skikd]	0.179005	0.433269	0.413	0.6795
wilaya[T.22 :sidi]	-0.806572	0.376631	-2.142	0.0322 *
wilaya[T.23 :annab]	-0.249626	0.395196	-0.632	0.5276
wilaya[T.28 :m'sil]	-0.375423	0.418248	-0.898	0.3694
wilaya[T.29 :masca]	-0.009983	0.420278	-0.024	0.9810
wilaya[T.31 :oran]	-0.722914	0.397283	-1.820	0.0688 .
wilaya[T.33 :illiz]	0.756495	0.612109	1.236	0.2165
wilaya[T.38 :tisse]	0.057158	0.424109	0.135	0.8928
wilaya[T.5 :batna]	-0.664330	0.380721	-1.745	0.0810 .
wilaya[T.7 :biskra]	-0.704509	0.388190	-1.815	0.0695 .
wilaya[T.9 :blida]	-0.051150	0.424555	-0.120	0.9041

Groupe9 2)A l'aide de package (vgam)sous R qui traite la variable de régression logistique polytomique

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.40635	0.06092	39.501	< 2e-16 ***
ptdjavec[T.2]	0.24689	0.17890	1.380	0.167591
ptdjavec[T.3]	15.17761	987.40581	0.015	0.987736
dejlieu[T.2]	0.80660	0.36728	2.196	0.028083 *
dejlieu[T.3]	0.84698	1.02020	0.830	0.406421
dejlieu[T.4]	1.39436	1.01308	1.376	0.168708
dinlieu[T.2]	1.19446	1.04002	1.148	0.250763
dinlieu[T.3]	15.42561	1920.09925	0.008	0.993590
dinlieu[T.4]	14.98994	5439.52703	0.003	0.997801
dinavec[T.2]	-0.88990	0.23548	-3.779	0.000157 ***
dinavec[T.3]	13.68669	1178.21305	0.012	0.990732

Après avoir éliminé les variables ayant un coefficient le moins significative l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	2.1697	0.3096	7.008	2.42e-12 ***
dejlieu	0.7198	0.2385	3.018	0.00255 **
dinavec	-0.4494	0.1822	-2.467	0.01363 *

Le modèle final pour la variable diabète selon les variables explicatives polytomiques. A l'aide de package (vgam)sous R qui traite la variable de régression logistique polytomique

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.28004	0.72349	4.534	5.8e-06 ***
actiprin[T.cadre mo]	-0.34066	0.49665	-0.686	0.492762
actiprin[T.cadre su]	-1.32129	0.81312	-1.625	0.104173
actiprin[T.employé]	0.43269	0.43560	0.993	0.320548
actiprin[T.enseigna]	0.16589	0.51217	0.324	0.746020
actiprin[T.femme au]	0.07961	0.35662	0.223	0.823363
actiprin[T.indéterm]	15.61337	1671.80250	0.009	0.992548
actiprin[T.ouvrier]	1.27270	0.42686	2.982	0.002868 **
actiprin[T.prof lib]	0.32203	0.81446	0.395	0.692552
actiprin[T.retraité]	-0.63512	0.35664	-1.781	0.074936 .
actiprin[T.sans pro]	0.49651	0.41215	1.205	0.228321
dejlieu[T.2]	0.27902	0.37185	0.750	0.453037
dejlieu[T.3]	0.81994	1.05069	0.780	0.435164
dejlieu[T.4]	1.04137	1.02772	1.013	0.310927
dinavec[T.2]	-0.61377	0.21222	-2.892	0.003826 **
dinavec[T.3]	15.02168	1207.30120	0.012	0.990073
nivinst[T.formatio]	0.42407	0.43120	0.983	0.325377
nivinst[T.moyen]	0.60311	0.21067	2.863	0.004198 **
nivinst[T.non préc]	-0.53396	1.16946	-0.457	0.647968
nivinst[T.primaire]	0.29404	0.14006	2.099	0.035790 *
nivinst[T.secondai]	1.03410	0.28028	3.690	0.000225 ***
nivinst[T.supérieu]	1.68256	0.62210	2.705	0.006837 **
smoke[T.Fumeur actuel]	0.36201	0.25610	1.414	0.157492
smoke[T.Non fumeur]	0.18601	0.20070	0.927	0.354038
situatm[T.divorcé]	-1.38538	0.61721	-2.245	0.024795 *
situatm[T.marié]	-1.22646	0.51967	-2.360	0.018272 *
situatm[T.non préc]	14.05165	3760.51086	0.004	0.997019
situatm[T.séparé]	-1.76054	0.92127	-1.911	0.056006 .
situatm[T.veuf]	-1.47040	0.53779	-2.734	0.006254 **
wilaya[T.12 :tebes]	0.56877	0.45691	1.245	0.213197
wilaya[T.15 :tizi]	0.11778	0.43750	0.269	0.787776
wilaya[T.16 :alger]	-0.52540	0.39055	-1.345	0.178530
wilaya[T.18 :jijel]	-0.18204	0.40843	-0.446	0.655806
wilaya[T.21 :skikd]	0.50253	0.44449	1.131	0.258230
wilaya[T.22 :sidi]	-0.49506	0.39001	-1.269	0.204310
wilaya[T.23 :annab]	-0.11830	0.40988	-0.289	0.772868
wilaya[T.28 :m'sil]	-0.13516	0.43049	-0.314	0.753549
wilaya[T.29 :masca]	0.10885	0.44044	0.247	0.804798
wilaya[T.31 :oran]	-0.43557	0.41543	-1.048	0.294410
wilaya[T.33 :illiz]	0.94586	0.62278	1.519	0.128821
wilaya[T.38 :tisse]	0.47302	0.43514	1.087	0.277009
wilaya[T.5 :batna]	-0.45325	0.39941	-1.135	0.256454
wilaya[T.7 :biskra]	-0.29692	0.40145	-0.740	0.459534
wilaya[T.9 :blida]	0.11031	0.43328	0.255	0.799042

Après avoir éliminé les variables ayant un coefficient le moins significative

l'une après l'autre et refaire la régression sans ces variables nous avons obtenu le tableau ci dessous

Coefficients	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.4816	0.6130	5.680	1.35e-08 ***
actiprin[T.cadre mo]	-0.2598	0.4882	-0.532	0.594687
actiprin[T.cadre su]	-1.3807	0.8191	-1.685	0.091895 .
actiprin[T.employé]	0.3785	0.4294	0.881	0.378126
actiprin[T.enseigna]	0.1208	0.5080	0.238	0.812087
actiprin[T.femme au]	-0.1088	0.3425	-0.318	0.750716
actiprin[T.indéterm]	15.5297	1552.0524	0.010	0.992017
actiprin[T.ouvrier]	1.1702	0.4168	2.807	0.004996 **
actiprin[T.prof lib]	0.3560	0.8017	0.444	0.656993
actiprin[T.retraité]	-0.8396	0.3501	-2.398	0.016480 *
actiprin[T.sans pro]	0.4246	0.3996	1.062	0.288056
dinavec[T.2]	-0.5709	0.2059	-2.773	0.005561 **
dinavec[T.3]	15.1383	1236.1287	0.012	0.990229
nivinst[T.formatio]	0.2314	0.4256	0.544	0.586677
nivinst[T.moyen]	0.4875	0.2038	2.393	0.016731 *
nivinst[T.non préc]	-0.8233	1.1455	-0.719	0.472330
nivinst[T.primaire]	0.2253	0.1358	1.659	0.097116 .
nivinst[T.secondai]	0.9122	0.2740	3.329	0.000873 ***
nivinst[T.supérieu]	1.5337	0.6273	2.445	0.014491 *
situatm[T.divorcé]	-1.4483	0.6131	-2.362	0.018159 *
situatm[T.marié]	-1.1234	0.5158	-2.178	0.029396 *
situatm[T.non préc]	14.1789	3785.4839	0.004	0.997011
situatm[T.séparé]	-1.7147	0.9153	-1.873	0.061009 .
situatm[T.veuf]	-1.3422	0.5340	-2.513	0.011955 *

arbre de décision

Arbre associé au sous échantillon 2 :

Résumé du modèle Arbre de décision pour Classification (construit avec 'rpart') : n= 700 node), split, n, loss, yval, (yprob) * denotes terminal node
1) root 700 58 0 (0.08285714 0.91714286)
2) age>=58.19712 151 30 0 (0.19867550 0.80132450)
4) sucre=0 42 18 0 (0.42857143 0.57142857)
8) age< 68.8679 34 16 1 (0.52941176 0.47058824)
16) saladver=1 12 3 1 (0.75000000 0.25000000) *
17) saladver=0 22 9 0 (0.40909091 0.59090909) *
9) age>=68.8679 8 0 0 (0.00000000 1.00000000) *
5) sucre=1 109 12 0 (0.11009174 0.88990826)
10) dessertjc< 0.5 27 5 0 (0.18518519 0.81481481) *
11) dessertjc>=0.5 28 0 0 (0.00000000 1.00000000) *
3) age< 58.19712 549 28 0 (0.05100182 0.94899818)

6) age>=38.17248 451 28 0 (0.06208426 0.93791574)
12) sucrette=1 7 3 0 (0.42857143 0.57142857) *
13) sucrette=0 443 25 0 (0.05643341 0.94356659)
26) boissons< 1060 399 25 0 (0.06265664 0.93734336)
52) boissons< 10 65 8 0 (0.12307692 0.87692308)
104) legumejc>=0.5 27 4 0 (0.14814815 0.85185185) *
105) legumejc< 0.5 14 0 0 (0.00000000 1.00000000) *
53) boissons>=10 334 17 0 (0.05089820 0.94910180) *
27) boissons>=1060 44 0 0 (0.00000000 1.00000000) *
7) age< 38.17248 98 0 0 (0.00000000 1.00000000) *

Arbre associé au sous échantillon 3 :

Résumé du modèle Arbre de décision pour Classification (construit avec 'rpart') :
n= 700

node), split, n, loss, yval, (yprob) * denotes terminal node
1) root 700 58 0 (0.08285714 0.91714286)
2) age>=58.19712 151 30 0 (0.19867550 0.80132450)
4) sucre=0 42 18 0 (0.42857143 0.57142857)
8) age< 68.8679 34 16 1 (0.52941176 0.47058824)
16) saladver=1 12 3 1 (0.75000000 0.25000000) *
17) saladver=0 22 9 0 (0.40909091 0.59090909) *
9) age>=68.8679 8 0 0 (0.00000000 1.00000000) *
5) sucre=1 109 12 0 (0.11009174 0.88990826)
10) dessertjc< 0.5 27 5 0 (0.18518519 0.81481481) *
11) dessertjc>=0.5 28 0 0 (0.00000000 1.00000000) *
3) age< 58.19712 549 28 0 (0.05100182 0.94899818)
6) age>=38.17248 451 28 0 (0.06208426 0.93791574)
12) sucrette=1 7 3 0 (0.42857143 0.57142857) *
13) sucrette=0 443 25 0 (0.05643341 0.94356659)
26) boissons< 1060 399 25 0 (0.06265664 0.93734336)
52) boissons< 10 65 8 0 (0.12307692 0.87692308)
104) legumejc>=0.5 27 4 0 (0.14814815 0.85185185) *
105) legumejc< 0.5 14 0 0 (0.00000000 1.00000000) *

53) boissons>=10 334 17 0 (0.05089820 0.94910180) *
27) boissons>=1060 44 0 0 (0.00000000 1.00000000) *
7) age< 38.17248 98 0 0 (0.00000000 1.00000000) *

Arbre associé au sous échantillon 4 :

1) root 700 56 0 (0.08000000 0.92000000)

2) age>=47.06639 407 51 0 (0.12530713 0.87469287)
4) sucre=0 84 28 0 (0.33333333 0.66666667)
8) dessertjc< 0.5 40 18 0 (0.45000000 0.55000000)

16) nbpieces>=3.5 12 2 1 (0.83333333 0.16666667) *
17) nbpieces< 3.5 28 8 0 (0.28571429 0.71428571)
34) saladver=1 11 5 1 (0.54545455 0.45454545) *
35) saladver=0 17 2 0 (0.11764706 0.88235294) *
9) dessertjc>=0.5 21 2 0 (0.09523810 0.90476190) *
5) sucre=1 323 23 0 (0.07120743 0.92879257)
10) milieu=urbain 206 20 0 (0.09708738 0.90291262)
20) nbpieces>=1.5 175 18 0 (0.10285714 0.89714286) *
21) nbpieces< 1.5 21 0 0 (0.00000000 1.00000000) *
11) milieu=rural 117 3 0 (0.02564103 0.97435897) *
3) age< 47.06639 293 5 0 (0.01706485 0.98293515) *

Arbre associé au sous échantillon 5 :

Résumé du modèle Arbre de décision pour Classification (construit avec 'rpart') :
n= 700
node), split, n, loss, yval, (yprob)
* denotes terminal node
1) root 700 58 0 (0.08285714 0.91714286)
2) age>=58.19712 151 30 0 (0.19867550 0.80132450)
4) sucre=0 42 18 0 (0.42857143 0.57142857)
8) age< 68.8679 34 16 1 (0.52941176 0.47058824)
16) saladver=1 12 3 1 (0.75000000 0.25000000) *
17) saladver=0 22 9 0 (0.40909091 0.59090909) *
9) age>=68.8679 8 0 0 (0.00000000 1.00000000) *
5) sucre=1 109 12 0 (0.11009174 0.88990826)
10) dessertjc< 0.5 27 5 0 (0.18518519 0.81481481) *
11) dessertjc>=0.5 28 0 0 (0.00000000 1.00000000) *
3) age< 58.19712 549 28 0 (0.05100182 0.94899818) 6) age>=38.17248 451 28 0 (0.06208426 0.93791574)
12) sucrette=1 7 3 0 (0.42857143 0.57142857) *
13) sucrette=0 443 25 0 (0.05643341 0.94356659)
26) boissons< 1060 399 25 0 (0.06265664 0.93734336)
52) boissons< 10 65 8 0 (0.12307692 0.87692308)
104) legumejc>=0.5 27 4 0 (0.14814815 0.85185185) *
105) legumejc< 0.5 14 0 0 (0.00000000 1.00000000) *

53) boissons>=10 334 17 0 (0.05089820 0.94910180) *

27) boissons>=1060 44 0 0 (0.00000000 1.00000000) *
7) age< 38.17248 98 0 0 (0.00000000 1.00000000) *

Arbre associé à l'échantillon global :

Résumé du modèle Arbre de décision pour Classification (construit avec 'rpart') :
n= 3372

node), split, n, loss, yval, (yprob) * denotes terminal node
1) root 3372 264 0 (0.078291815 0.921708185)
2) sucre=0 730 138 0 (0.189041096 0.810958904)
4) age>=50.95961 354 110 0 (0.310734463 0.689265537)
8) confitur=0 335 110 0 (0.328358209 0.671641791)
16) saladver=1 135 61 0 (0.451851852 0.548148148)
32) limonade=0 95 43 1 (0.547368421 0.452631579)
64) pomme=1 16 2 1 (0.875000000 0.125000000) *
65) pomme=0 79 38 0 (0.481012658 0.518987342)
130) age>=60.99247 41 16 1 (0.609756098 0.390243902)
260) boissons>=375 16 3 1 (0.812500000 0.187500000) *
261) boissons< 375 25 12 0 (0.480000000 0.520000000)
522) age< 64.86926 10 2 1 (0.800000000 0.200000000) *
523) age>=64.86926 15 4 0 (0.266666667 0.733333333) *
131) age< 60.99247 38 13 0 (0.342105263 0.657894737) *
33) limonade=1 40 9 0 (0.225000000 0.775000000) *
17) saladver=0 199 49 0 (0.246231156 0.753768844)
34) boissons>=550 46 18 0 (0.391304348 0.608695652) *
35) boissons< 550 153 31 0 (0.202614379 0.797385621)
70) peche=1 20 8 0 (0.400000000 0.600000000) *
71) peche=0 133 23 0 (0.172932331 0.827067669)
142) mouton=1 10 5 1 (0.500000000 0.500000000) *
143) mouton=0 123 18 0 (0.146341463 0.853658537)
286) legumejc>=1.5 15 5 0 (0.333333333 0.666666667) *
287) legumejc< 1.5 35 2 0 (0.057142857 0.942857143) *
9) confitur=1 19 0 0 (0.000000000 1.000000000) *
5) age< 50.95961 376 28 0 (0.074468085 0.925531915)
10) dessertjc< 0.5 101 19 0 (0.188118812 0.811881188) *
11) dessertjc>=0.5 97 1 0 (0.010309278 0.989690722) *
3) sucre=1 2640 126 0 (0.047727273 0.952272727)
6) age>=48.14237 1384 102 0 (0.073699422 0.926300578)
12) milieu=urbain 860 79 0 (0.091860465 0.908139535)
24) dessertjc< 0.5 186 24 0 (0.129032258 0.870967742) *
25) dessertjc>=0.5 263 15 0 (0.057034221 0.942965779) *
13) milieu=rural 524 23 0 (0.043893130 0.956106870) *
7) age< 48.14237 1256 24 0 (0.019108280 0.980891720)
14) age>=38.16975 902 24 0 (0.026607539 0.973392461)
28) age< 38.1985 7 2 0 (0.285714286 0.714285714) *
29) age>=38.1985 895 22 0 (0.024581006 0.975418994)
58) couscou1jc< 0.5 172 9 0 (0.052325581 0.947674419) *
59) couscou1jc>=0.5 120 1 0 (0.008333333 0.991666667) *
15) age< 38.16975 354 0 0 (0.000000000 1.000000000) *

Arbre de décision sur les variables polytomies :

1) root 700 53.1942900 1.917143
2) wilaya=1 :adrar,16 :alger,22 :sidi,5 :batna,7 :biskra,9 :blida 316 35.6803800 1.870253

4) actiprin=enseigna,femme au,retraité,sans pro 216 31.9583300 1.819444
8) nivinst=analphab,moyen,primaire 187 30.2780700 1.796791
16) situatm=veuf 27 6.0000000 1.666667
32) wilaya=16 :alger,5 :batna 12 3.0000000 1.500000 *
33) wilaya=1 :adrar,22 :sidi,7 :biskra,9 :blida 15 2.4000000 1
.800000 *
17) situatm=célibata,divorcé,marié,séparé 160 23.7437500 1.818750

34) wilaya=1 :adrar,22 :sidi,7 :biskra 66 12.6212100 1.742424 *

35) wilaya=16 :alger,5 :batna,9 :blida 94 10.4680900 1.872340 *

9) nivinst=formatio,secondai,supérieu 29 0.9655172 1.965517 *
5) actiprin=artisan,cadre mo,cadre su,employé,indéterm,ouvrier,prof lib 99 1.9595960 1.979798 *

BIBLIOGRAPHIE

[1] projet . TAHINA *: Transition épidémiologique et système de santé* (Contrat n ICA3-CT-2002-10011) 2005

[2] .Lebart. *: traitement statistique des enquêtes*

[3] Hosmer DW and Lemeshow S. *: Applied logistic regression.* John Wiley Son 1989»

[4] Amemiya T *: Advanced Econometrics. Cambridge, Harvard University Press* (2000)

[5] T . *Econométrie des Variables Qualitatives. Dunod,*(2000)

[6] Yves Lechevallier. *Méthodes de classification supervisées, Les méthodes de segmentation ou les arbres de décision*(1985).

[7] Breiman al. *: Classification and Régression Tree.* Wadsworth, Belmont(1984)

[8] Assadi Réza Khattar Karim. *: L'utilisation d'un réseau de neurones pour optimiser la gestion d'un firewall.* École Polytechnique de Montréal. Québec, Canada.(1 mai 2002)

[9]] Benahmed Nadia. *: Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isoles : sélection et pondération des primitives par l'algorithmes Génétiques.* Ecole de technologie supérieure. Université de Québec. Canada.(Mars 2002)

[10] Bishop Christopher M. *: Neural networks for pattern récognition.* Birmingham : Clarendon press Oxford(1995).

[11] Claude TOUZET. *: Réseaux de neurones artificiels a la robotique coopérative.* Université de Droit, d'Economie et des Sciences

d'Aix-Marseille III Faculté des Sciences et Techniques de Saint-Jérôme$^{F}rance(Septembre1998))$

[12] Crucianu Mihail. *: Algorithmes d'évolution pour les réseaux de neurones.* Laboratoire d'Informatique ; Ecole d'Ingénieurs en Informatique pour l'industrie (France)(Avril 1997)

[13] Dreyfus G., Martinez J.M., Samueliesc M *:Réseaux de neurones méthodologie et application.* Edition Eyrolles(2004).

[14] F-C.chen . *: Back-propagation neural network for nonlinear self-tuning adaptative control, IEEE control System* Magazine T-page(41-48)(1990)

[15] Guyon I, Poujaud I., Personnaz L., Dreyfus G., Denker J., Le Cun Y. *: Comparing différent neural networks architectures for classifying handwritten digits2.* Int. J. Conf. on Neural Networks, 2, 127-132. Washington, USA (1989).

[16] Jodouin Jean-François . *: Les réseaux neuromimétiques : modèles et applications.* Édition Hermès (1994).

[17] Jodouin Jean-François . *: Les réseaux de neurones : principes et définitions. Edition Hermès(1994).*

[18] Kary FRÂMLING. :Les réseaux de neurones comme outils d'aide à la décision floue Ecole Nationale Supérieure des Mines de Saint-Etienne. France (Juillet 1992).

[19] Krranowsky.w.j. : Dscrimination and Classification Using Both binary tree hypothesis (25-27.Aprill995).

[20] L.L., Alin Morineau, M.P. : Statistique exploratoire multidimensionnelle. Paris (juilletl995).

[21] Lerman.I.C Ph. Peter, : Elaboration et logiciel d'un indice de similarité entre objets d'un type quelconque. Application au problème du consensus en classification. Publication Interne Irisa n.262, Rennes (juillet 1985).

[22] Medsker Larry, Liebowitz Jay . : Design and Development of Expert Systems and Neural Networks. Édition MacMillan Publishing Company(1994).

[23] Morgan. J.N J.A. Sonquist. :Problems in analysis of survy data and a prosal. J. Am. Statist.Assoc,58,pp.415-434.(1963).

[24] O.Kinouchi.M.H.R. : Tragtenberg, « Modeling neurons by simple maps » international journal of bifurcation and chos, vol.6, NI 2 A,pp. 150-1560 (1996).

[25] Rem m Jean-François. :Probabilités et réseaux de neurones. CRIN-CNRS et INRIA-Lorraine Rapport Technique RR n2909.

[26] Rival Isabelle. : LES Réseaux de Neurones Formels pour le pilotage de Robots Mobiles, Laboratoire d'Électronique de l'ESPCI (École Supérieure de Physique et de Chimie Industrielles).

[27] Rivais. I, Personnaz L., Dreyfus G., Ploix J.L. : Modélisation, classification et commande par réseaux de neurones : principes fondamentaux, méthodologie de conception, et illustrations industrielles , in Récents progrès en génie des procédés 9, Lavoisier technique et documentation, Paris.(1995).

[28] Rosenblatt,R. :Principles of Neurodynamics. Spartan Books.New-York

[29] S.Haykin. : Neuronal Network. A comprehensive fondation, Macmillan, New York (1958)(1994)

[30] J Zapranis Achilleas, Refenes Apostolos-Paul . : Principles of Neural Model Identification, Sélection and Adequacy with applications to financial Econometrics. Edition Springer- Verlag London Berlin Heidelberg(1999).

[31] Zurada J.M. : Introduction to artificial neural Systems. Minnesota : West publishing company(1992).

[32] Fawcett.Receiver Operating Characteristics (2003)

[33] http ://fr.wikipedia.org