TP L3 ISIMA : Estimation II

Ce TP a pour objectif la création d'estimateurs, et en particulier de modèles de régression. On s'intéressera également à la comparaison de leur efficacité.

Présentation

Soit $(x_i, y_i)_{i=1}^{i=nb_{ex}}$ une base de données étiquettée : les x_i correspondent à des situations, les y_i à des réponses adaptées à ces situations. Le but de cette partie est de découvrir la création d'estimateurs, c'est à dire de déterminer un *modèle de régression* de X en Y où X est la variable aléatoire dont des réalisations sont les x_i et Y est la variable dont des réalisation sont les y_i. On s'intéresse ici à des modèles fonctionnels, c'est à dire que l'estimateur de Y sera une fonction de X. Voici quelques exemples :

Situation (x_i)	**Réponse attendue** (y_i)	**But du travail : estimer ...**
Poids d'un individu	Taille de cet individu	la taille connaissant le poids
Poids et tour de poignet	Taille de cet individu	la taille sachant le poids et le tour de poignet
Age de la maitresse	Nombre d'élèves de la classe	le nombre d'élèves en fonction de l'âge de la maitresse
Une image	Présence d'un chien (oui/non)	si un tableau de pixel contient une représentation de chien

On peut définir n'importe quel estimateur, depuis celui qui ne sert à rien (par exemple il répond toujours '42') jusqu'à celui qui ne se trompe jamais. Un point important est de définir la qualité d'un estimateur, cela sert à choisir parmi plusieurs estimateurs celui qu'il est préférable d'utiliser, et également de rechercher à améliorer un estimateur déjà existant.
Soit θ_n un estimateur de Θ fabriqué à partir de n réalisations. Les deux mesures de qualité les plus fréquentes d'un estimateur sont :

- $b(\theta_n) = E(\theta_n) - \Theta$ que l'on appelle le biais. Les cas sympathiques sont :
  - $b(\theta_n) = 0$, l'estimateur est dit non biaisé
  - $b(\theta_n) =$ constante, auquel cas on fabrique immédiatement $\kappa_n = \theta_n -$ la constante qui est alors non biaisé
  - $b(\theta_n)$ est une fonction dont la norme tend vers 0 lorsque n tend vers l'infini. On parle alors d'estimateur asymptotiquement non biaisé.
- $EQM(\theta_n) = E\left((\theta_n - \Theta)^2\right)$, que l'on appelle l'erreur quadratique moyenne (dont le nom est mal choisi, puisqu'il s'agit d'une espérance et non d'une moyenne). Souvent, on parlera également de l'erreur type, qui est la racine de l'erreur quadratique moyenne. Dans la suite de ce TP, on utilisera EQM pour parler ce cette mesure, et de 'moyenne de l'erreur quadratique' lorsqu'on parlera réellement de moyenne et non d'espérance.

Il existe quelques cas théoriques où le biais ainsi que l'erreur quadratiques sont bien définis par des résultats théoriques. Les deux premiers sont :

- Un estimateur de l'espérance de biais nul est... la moyenne
- Un estimateur non biaisé de la variance est $\frac{n}{n-1}\sigma^2$ où σ^2 est la variance sur n éléments tirés dans la population.

Lorsqu'on construit un estimateur, il n'est en général pas possible de calculer l'EQM, et on doit se contenter de l'approcher via une estimation de l'espérance sur un ensemble de réalisations.

Premiers pas

1 Deviner l'impossible ?

Dans le fichier 'TP3_etu.py', vous diposez d'un générateur 'impossible' (qui s'utilise comme dans le TP précédent) qui renvoie un entier tiré en uniforme dans $[1..N]$, ou N est inconnu. Le but de cet exercice est de réaliser plusieurs estimateurs de N, et de déterminer celui qui est le plus efficace.

1. Estimateur $\phi_n = 2 \times$ moyenne des réalisations $- 1$.
   (a) Présenter l'idée derrière cet estimateur
   (b) Estimer son biais ainsi que son erreur quadratique moyenne
2. Estimateur $\psi_n = 2 \times$ médiane des réalisations.
   (a) Présenter l'idée derrière cet estimateur
   (b) Estimer son biais ainsi que son erreur quadratique moyenne
3. Estimateur $\zeta_n = Max$(moyenne des réalisations).
   (a) Présenter l'idée derrière cet estimateur
   (b) Estimer son biais ainsi que son erreur quadratique moyenne
4. Construire un estimateur, si vous le désirez à partir des autres (combinaison linéaire, si-alors, ...)
   (a) Présenter l'idée derrière cet estimateur
   (b) Estimer son biais ainsi que son erreur quadratique moyenne
5. Conclure

2 Superficie d'un champ

Un champ est carré (on ne remettra pas cela en cause). Un satellite mesure à plusieurs reprises la longueur d'un côté du champ. On admet que la mesure renvoyée par le satellites suit une loi gaussienne dont l'espéance est la longueur de ce côté et l'écart type inconnu.

6. Le satellite renvoie une liste de mesures du côté du champ, proposer en fonction de ces mesures plusieurs estimations de la superficie du champ.
7. En simulant les mesures renvoyées par le satellite, déterminer la meilleure mesure, en prenant comme critères le biais ainsi que l'erreur quadratique moyenne

Création d'un modèle de régression

Soit $T = (x_i, y_i)_{i=1}^{i=nb_{ex}}$ une base de données étiquettée. La création d'un modèle de régression de X en Y peut suivre le plan suivant :

— Choix d'un ensemble de fonctions $\mathbb{F}$: le but de la suite sera de choisir la 'meilleure' fonction $f \in \mathbb{F}$ telle que $f(X) \approx Y$

— Séparer en deux la base de données : $T = T_A \cup T_V$. On utilisera T_A pour choisir la fonction f et T_V pour valider le choix réalisé. Conserver une partie de la base de données pour la validation va permettre d'estimer l'EQM. Si on calcule l'EQM sur les données qui ont permi de choisir f, notre EQM sera une moyenne de l'erreur quadratique sur les situations rencontrées, et cela ne présagera en rien sur la qualité de f sur des situations non connues ; on appelle ce problème le *sur-apprentissage* ou *overfitting*.

— Choisir $f \in \mathbb{F}$ qui minimise

$$eqm(f) = \frac{1}{n} \sum_{i=1}^{|T_A|} (f(x_i) - y_i)^2$$

— Estimer l'erreur quadratique moyenne : pour cela on calcule

$$eqm(f) = \frac{1}{n}\sum_{i=1}^{|T_V|}(f(x_i) - y_i)^2$$

et on espère :-) que l'EQM est assez proche de cette valeur (pour faire mieux, Cf. TP précédent)

3 Première mise en oeuvre

Afin d'accélérer votre progression, vous disposez d'un code déjà écrit 'TP3_etu.py' qu'il va vous falloir compléter.

1. Générer des données sur la droite d'équation $y = -5x + 2$, puis perturber très légèrement les valeurs des ordonnées par une variable aléatoire gaussienne. Pour cela, compléter le code de 'genere_droite_bruitee'
2. Séparer en deux ensembles l'ensemble des points ainsi construits : écrire une fonction 'separe' qui prend une liste 'L' en entrée et qui renvoie un couple de listes $(L_A,\ L_V)$, vérifiant les conditions : $\left|\begin{array}{l} L_A \cap L_B = \emptyset \\ L_A \cup L_B = L \\ \forall x \in L, P(x \in L_A) = \frac{1}{2} \end{array}\right.$
3. Définir la fonction modèle, cette fonction prend en entrée une situation (un tuple de floats) ainsi qu'un tuple de paramètres, et renvoie l'évauation de la fonction (précisée par les paramètres) au point choisi. Pour cela compléter le code de la fonction 'modele_1'
4. La fonction principale effectue le travail suivant :
   — Définir le nombre de données (une constante au début de la fonction)
   — Créer lees données (utiliser 'genere_droite_bruitee')
   — Choisir le modèle (ici 'modele_1')
   — Déterminer un tuple minimal ainsi qu'un tuple maximal pour la position des graines des algorithmes de recherche de paramètres
   — Séparer les donnéees en deux : L_A et L_V (utiliser 'separe')
   — Sur L_A, utiliser la fonction 'recherche_param' (fournie) afin de déterminer des paramètres qui minimisent la moyenne des erreurs quadratiques du modèle sur L_A
   — Comme on est en dimension 2, faire une représentation graphique présentant les points et le modèle
   — Calculer une estimation de l'EQM, en calculant la moyenne de l'erreur quadratique sur L_V
5. Créer la fonction 'modele_polynome_reel' qui prend en entrée un tuple contenant un float 'x' et un tuple $(a_0, a_1, ..., a_n)$ et qui renvoie $\sum_{k=0}^{n} a_k x^k$
6. Recommencer le travail en prenant successivement comme modèle une parabole, un polynôme de degré 3, 4, 5, ...
7. Construire une fonction qui renvoie le modèle le meilleur parmi les polynômes de degré 0, 1, 2, ... pour la valeur maximale du degré du polynôme à tester, déterminer une condition d'arret raisonnable.
8. Représenter l'évolution de la moyenne des erreurs quadratiques sur L_A et sur L_V (deux courbes sur le même graphique) en fonction de la complexité du modèle
9. Expliquer ce qui se passe (des graphiques sont bienvenus dans les explications). Attention : bien faire la différence entre les insuffisances de la fonction de minimisation et le sur-apprentissage

4 On passe à deux variables

1. Ecrire une fonction 'genere_deux_variables' qui renvoie une liste de 1200 couples dont le premier élément est un couple $(x_1, x_2) \in [-42\,;\,42] \times [0\,;\,7]$ et le second est une valeur $y = 13 \times x_1 + 3 \times x_1 x_2 - 5 \times x_2 - 27 +$ bruit, où bruit est la réalisation d'une variable aléatoire suivant une loi normale centrée d'écart type 150
2. Ecrire une fonction 'modele_deux_variables' prenant comme entrées :
   — x un couple de floats (x_1, x_2)
   — param un quadruplet (a, b, c, d)

et qui renvoie $a \times x_1 + b \times x_1x_2 + c \times x_2 + d$

3. En suivant le plan de l'exercice précédent, déterminer les paramètres du modèle, et estimer son EQM.

Construction de modèles à partir de bases de données

Le fichier fourni dispose d'une fonction 'recupere_csv' que vous pouvez adapter à vos besoins.
Pour construire un modèle :

— Vous sortez une famille F de modèles de votre chapeau

— Vous séparez vos données en deux (L_A, L_V)

— Vous choisisez l'élément de la famille F qui minimise la moyenne des erreurs quadratiques sur L_A

— Vous estimez l'EQM sur L_V et vous en déduisez une estimation de l'erreur type

— Si cette erreur type vous semble suffisament faible, vous livrez la fonction

— Sinon vous choisissez une autre famille de fonctions et vous recommencez... ou vous convainquez votre client qu'avec les données fournies, on ne peut pas faire mieux.

5 Consommation de carburant

Le fichier 'Modélisation_Conso_Gasoil.csv' contient un relevé de différentes informations de la consommation du carburant d'un poids lourd.

4. Créer une fonction prenant en entrée le poids du chargement ainsi que le nombre de kilomètres à rouler et qui renvoie une estimation de la consommation de carburant pour ce trajet.

6 Freinage

Le fichier 'Modélisation_Freinage.csv' contient un relevé de couples (vitesse, distance de freinage).

5. Votre travail consiste à créer une fonction prenant en entrée une vitesse exprimée en km/h qui renvoie une estimation de la distance de freinage exprimée en mètres.

7 Modélisation tension

Le fichier 'Modélisation_Tensions.csv' contient deux colonnes, qui sont des tensions mesurées en deux points d'une carte électronique (qui fonctionne).

6. Le travail consiste à construire un modèle de la tension2 (en volts) en fonction de la tension1 (en volts).

8 Plus compliqué... mais plus abouti

Le but de cette étude est de construire un processus de test de composants électronique qui soit économique sans trop de perte de qualité. Plus précisément, avant l'étude, les pièces étaient produites puis contrôlées une à une. Ce procédé avait l'avantage d'une grande fiabilité au détriment d'un coût très important. Vous êtes parvenu à diminuer grandement le côut de fabriaction, et vous voudriez diminuer maintenant le coût du controle. La procédure que vous désirez suivre est la suivante :

— Mesurer la tension en deux points de la carte (que vous a indiqué Monsieur Bob, qui travaille à la chaîne de contrôle depuis 40 ans)

— Selon les valeurs mesurées, décider :

  — S'il faut envoyer la pièce chez le client,

  — S'il faut mettre la pièce au rebut

  — S'il faut transmettre la pièce à Monsieur Bob afin qu'il la controle de façon exhaustive

Dans le fichier 'Modélisation_tests.csv' vous disposez dans les deux premières colonnes des mesures de tensions, dans la troisième colonne d'un indicateur valant '1' si la carte fonctionne et '0' sinon.

7. Réaliser un modèle prenant en entrée les deux tensions et renvoyant un estimateur du fonctionnement de la carte. Voici quelques pistes pour construire votre modèle :
   (a) Pour les cartes qui fonctionnent, réaliser un modèle de Tension2 f en fonction de Tension1. Définir deux seuils $seuil_{bas}, seuil_{haut}$. La règle de décision est alors : répondre 'fonctionne' ssi
   $$seuil_{bas} \leq f(tension1) - tension2 \leq seuil_{haut}$$
   (b) Prendre N points au hasard (ou bien choisis...) dans l'espace des couples possibles (tension1, tension2), que l'on appelle des 'situations de référence'. Affecter à chaque 'situation de référence' une valeur 'fonctionne' ou 'ne fonctionne pas'. La réponse est alors celle de la situation de référence la plus proche.
   (c) knn (k-nearest-neighbors) : Repondre de la même façon que la majorité des k voisins connus de la base L_A.
   (d) ...
8. On crée une troisième valeur de sortie à l'estimateur, '0.5', d'interprétation 'Je ne sais pas'. Comment modifier la formule du calcul de l'erreur de façon à ce qu'il n'y ait pas un nombre démesuré de 'je ne sais pas' (auquel cas votre travail est parfaitement inutile), et que pourtant le nombre d'erreurs de classification soit négligeables.
9. On désire maintenant réaliser des décisions en fonction des profits et coûts engendrés par nos erreurs de classification. Après analyse des coûts, on arrive à la conclusion :
   — 4% des pièces produites sont défectueuses
   — Un client content rapporte 20 euros
   — Une pièce défectueuse coûte 60 euros
   — Un contrôle complet coûte 10 euros
   — Un client mécontent coûte 130 euros (ce qui comprend le bénéfice attendu, le coût de la pièce défecteuse envoyée en premier, la gestion du litige, l'envoi d'une pièce dont on a vérifié totalement la qualité, et la perte d'image)
10. Si on contrôle toutes les pièces, calculer l'espérance du gain par pièce (papier/crayon)
11. Si on n'effectue aucun contrôle, calculer l'espérance du gain par pièce (papier/crayon)
12. Définir un modèle permettant de maximiser les bénéfices
13. Reprendre la question précédente, mais avec des variables pour les différents coûts et probabilités

---

Achat d'un appartement à Paris

On suppose que la qualité d'un appartement peut être mesurée par un réel. On définit Q la variable aléatoire qui associe à un appartement dans votre gamme de prix sa qualité. On suppose que Q suit une loi normale d'espérance et d'écart type inconnus. La situation est la suivante :

— Vous allez visiter un maximum de 122 appartements,
— Une visite vous donne la connaissance d'une réalisation de Q
— Après chaque visite, vous décidez si vous achetez ou pas l'appartement
— Si vous n'avez pas acheté après la 121-ième visite, vous acheterez le 122-ième appartement

14. Avant de traiter le problème, on va s'intéresser à un problème simplifié. On connait l'espérance et l'écart type de Q (vous êtes vendeur en immobilier à Paris, vous connaissez parfaitement le marché, et vous cherchez un bien pour votre propre compte). Ecrire une fonction qui décide si vous devez acheter.
15. Retour au problème complet : créer une fonction qui décide si vous devez acheter, votre but étant de maximiser la qualité de votre achat final.