Descente de gradient & régression

Vincent Guigue

Exercice 1 – Régression simple et indicateurs statistiques

Une entreprise veut analyser ses coûts de production de son produit principal et en particulier les décomposer en coûts fixes et coûts variables et vérifier si ceux-ci sont, ou non, proportionnels aux quantités produites.

Elle postule donc un modèle linéaire $Y = \alpha + \beta X + \varepsilon$ où : X est la quantité produite (en milliers d'unités) ; Y le coût de production total (en milliers d'euros) ; β est le coût marginal de production (= coût nécessaire pour produire une unité supplémentaire) ; α représente les coûts fixes ; et ε est le résidu aléatoire.

Il dispose de données sur les $n = 10$ derniers mois :

$mois_i$	1	2	3	4	5	6	7	8	9	10
x_i	100	125	175	200	500	300	250	400	475	425
y_i	2 000	2 500	2 500	3 000	7 500	4 500	4 000	5 000	6 500	6 000

Q 1.1 Calculer les moyennes empiriques $\bar{x}$ et $\bar{y}$, les écarts-types empiriques s_x et s_y, la covariance empirique $cov(x, y)$ et le coefficient de corrélation linéaire r.

Q 1.2 Retrouver les expressions de α et β en calculant l'espérance de Y puis la covariance de X, Y.
Estimer a et b en fonction de ces expressions. Exprimer b en fonction de r.

Exercice 2 – Régression linéaire

Q 2.1 Régression linéaire 1D
Nous disposons d'un ensemble de N données $\{(x_i, y_i)_{i=1,\dots,N}\}$ à partir duquel nous souhaitons apprendre une droite de régression de Y sur X. Notre estimateur aura donc la forme suivante : $\hat{y}_i = f(x_i) = ax_i + b$. Notre but est de trouver les meilleurs coefficients a et b.
Pour une droite donnée l'erreur de régression cumulée au sens des moindres carrés est déterminée par $\sum_i e_i^2$ où $e_i = f(x_i) - y_i$, et $f(x_i)$ est l'ordonnée du point d'abscisse x_i.

Notons X la matrice $N \times 2$ des entrées avec ajout d'une colonne de termes constants : $X = \begin{bmatrix} x_1 & 1 \\ \vdots & \\ x_N & 1 \end{bmatrix}$ et $Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$.

Notons $D = \begin{bmatrix} a \\ b \end{bmatrix}$ le vecteur des paramètres de la droite de régression.

Q 2.1.1 Montrer que l'ensemble des estimations pour les entrées X peut être calculé matriciellement en utilisant la formule suivante : $\hat{Y} = XD$ (vérifier les dimensions et détailler le calcul d'une ligne).

Q 2.1.2 Montrer que l'erreur cumulée est calculée matriciellement en utilisant la formule suivante : $E = (XD - Y)^t(XD - Y)$
NB : dans un premier temps, détailler les dimensions de chaque matrice et calculer sur la dimension de E. Développer ensuite la formulation A^tA à l'aide d'une somme pour revenir à la formulation classique de l'erreur.

Q 2.1.3 Une fois le critère d'erreur E établi, quel problème d'optimisation devons nous résoudre pour trouver la droite de régression optimale ?
NB : nous sommes dans un cadre convexe : la fonction E de paramètres D admet un seul optimum global qui est un minimum. Rappeler la manière de trouver un optimum.

Q 2.1.4 Montrer que la dérivée de l'erreur, par rapport à D, s'écrit sous la forme matricielle suivante : $\nabla_D E = 2X^t(XD - Y)$
NB : détailler le calcul de chaque dérivée partielle et refactoriser pour obtenir la forme matricielle.

Q 2.1.5 Calculer les paramètres optimaux en résolvant analytiquement le problème sous forme matricielle.

Q 2.1.6 Simplifions temporairement le problème en considérant un biais nul. Quelle est la forme de la fonction $E(a)$? Tracer sommairement $E(a)$. Quelles sont les propriétés de $E(a)$ (combien de minimum...) ?

Q 2.2 Algorithme itératif pour la régression linéaire
Dans le cas général, on cherche à optimiser une fonction continue dérivable $C(W)$ d'un vecteur de paramètres W. Pour résoudre un tel problème, on peut utiliser un algorithme de gradient comme celui proposé ci-dessous :

Algorithme 1 : Descente de gradient

Initialisation des W;
$t = 1$;
repeat
  $W_{t+1} = W_t - \varepsilon \frac{\partial C}{\partial W}$;
  $t = t + 1$;
until ($C(W)$ n'évolue plus);

Q 2.2.1 Quel est l'intérêt d'utiliser un algorithme itératif (dont la solution est une approximation du point optimal) alors que nous disposons d'une solution analytique ?

Q 2.2.2 Adapter l'algorithme de descente de gradient pour la régression linéaire.

Q 2.2.3 L'algorithme est initialisé avec les paramètres suivants : $D^0 = (b^0, a^0)$. Si a^0 est plus grand que le $a^\star$ optimal, le gradient $\frac{\partial E}{\partial a}$ est-il positif ou négatif ? Idem si a^0 est plus petit que le a optimal.
Ces résultats vous semblent-ils cohérents avec l'algorithme de descente de gradient ?

Q 2.2.4 Même question avec b^0.

Q 2.2.5 Imaginez ce qui se passe si l'on optimise uniquement par rapport à a, en supposant que la valeur optimale de b est connue, pour différentes valeurs d'ε (valeur très grande, valeur très petite) : l'algorithme précédent peut-il diverger ou converge-t-il toujours vers la bonne solution ?

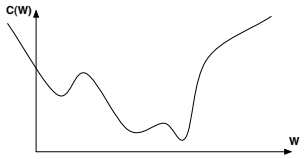

FIGURE 1 – Exemple de fonction coût $C(W)$ non-convexe

Q 2.2.6 Nous avons utilisé jusqu'ici des estimateurs linéaires. Donner un exemple d'estimateur non linéaire pour la régression 1D. Dans le cas non-linéaire, la fonction $C(W)$ est parfois non convexe (cf figure 1).

Q 2.2.7 Que pensez-vous de l'algorithme de gradient discuté précédemment dans ce cas ? L'algorithme de gradient converge-t-il toujours ? vers la solution optimale ?

Exercice 3 – Regression(s)

Soit un ensemble de N couples de valeurs tirées dans $\mathbb{R}^2$ de manière i.i.d. : $\{(x_i, y_i)\}_{i=1,\dots,N}$. L'enjeu de la régression est de prédire Y à partir de X.
Par rapport au nuage de points considéré (Fig. ci-contre), un statisticien vous recommande un modèle quadratique simple (représenté en ligne continue sur la figure). L'hypothèse est alors la suivante :

$$Y = \alpha X^2 + \beta + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma)$$

Comme dans le cours, l'idée est que les données suivent une distribution Gaussienne autour de $\alpha X^2 + \beta$. Ainsi, la vraisemblance d'une observation est donnée par :

$$p(Y|X, \alpha, \beta, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}\left(Y - (\alpha X^2 + \beta)\right)^2\right)$$

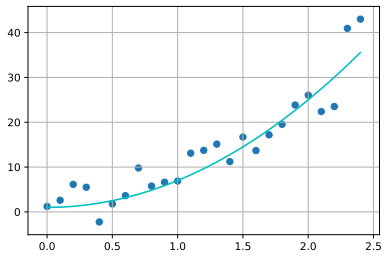

Fig. Données mesurées et modèle quadratique appris sur ces données.

Q 3.1 Dans l'optique d'estimer les paramètres du modèle quadratique, donner la formulation de la log-vraisemblance de cet échantillon.

Q 3.2 Montrer que l'optimisation de la vraisemblance par rapport à α et β mène à un système de deux équations linéaires à deux inconnues.

Q 3.3 Ce système s'écrit sous la forme matricielle : $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} e \\ f \end{bmatrix}$, soit : $A \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = B$
Donner les valeurs de a, b, c, d, e, f.

Q 3.4 Donner le code de la fonction `maxvraisemblance` qui prend en argument les vecteurs `x` et `y` contenant les données et qui retourne α et β.
Note : on ne s'occupe pas des imports et la fonction qui résout le système est `numpy.linalg.solve(A, B)`

Q 3.5 Une fois les valeurs de α et β trouvées, estimer le niveau de bruit σ dans les données en annulant la dérivée de la vraisemblance par rapport à σ. Sur quelle formule retombez-vous ?

Un expert du domaine vous explique que ces données sont en réalité issues de deux systèmes opérant en parallèle : l'un des systèmes est linéaire $Y = \alpha_1 X + \beta_1 + \varepsilon$ et l'autre du troisième degré $Y = \alpha_2 X^3 + \beta_2 + \varepsilon$
L'expert ajoute que les deux systèmes sont équi-probables pour la génération des observations.
Note : les ε suivent toujours une loi normale et le niveau de bruit est le même pour les deux modèles.

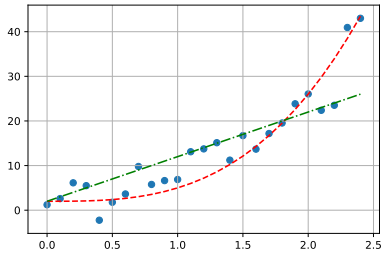

Fig. Données mesurées et mixture de modèles (linéaire & polynome d'ordre 3) appris sur ces données.

Q 3.6 Quelle approche peut vous permettre d'estimer tous les paramètres des deux modèles ? Décrire en quelques lignes les principales étapes et les pré-requis pour cette approche.

Q 3.7 Soit des paramètres initiaux $\theta_1 = \{\alpha_1^0, \beta_1^0\}, \theta_2 = \{\alpha_2^0, \beta_2^0\}$, donner l'expression des $Q_i^0(\theta)$. Rappeler la taille de cette matrice.

Q 3.8 On rappelle que la log-vraisemblance s'écrit ensuite : $\log \mathcal{L} = \sum_i \sum_j Q_i^0(\theta_j) \log\left(\frac{p(y_i, \theta_j)}{Q_i^0(\theta_j)}\right)$
Dériver la log-vraisemblance prédéfinie par rapport à α_1 à Q^0 constants et simplifier l'équation. Comment interpréter le résultat ?

Q 3.9 Pour l'implémentation, on envisage une simplification de l'approche en affectant en dur chaque point au modèle le plus probable. Donner le code de la fonction `maxvraisemblance_2` qui prend en argument les `x` et `y` contenant les données et qui retourne α_1, α_2, β_1 et β_2. L'ensemble de la procédure sera codé dans la fonction.
Note : pour simplifier, on considèrera arbitrairement $\sigma = 1$ [ça ne change rien pour l'affectation des points aux modèles].
Note : il est nécessaire de calculer les deux systèmes d'équations linéaires correspondant aux deux modèles... Mais leur forme est quasi identique à celle de la question Q 3.3 et on ne sera pas très sévère sur cet aspect.

Q 3.10 Ces données jouets ont été tirées aléatoirement... Néanmoins, imaginez un problème réel qui airait pu mener à ce tirage : expliquer simplement à quoi correspondent les axes des absisses et ordonnées dans ce cas.