Cost function

Michel Donnet

January 10, 2024

Contents

1 Fonction de coût pour la régression logistique

Afin d'entraîner les paramètres de la régression logistique, il faut pouvoir comparer les résultats obtenus par la régression avec les résultats attendus.

On souhaite définir une fonction à minimiser permettant de trouver les paramètres optimaux de la régression logistique.

Notre classification se base sur la fonction sigmoïde $\sigma(z) = \frac{1}{1+e^{-z}}$.

Comme la fonction exponnentielle est toujours positive, on a bien que $\sigma(z) \in [0, 1]$.

La fonction sigmoïde nous donne la probabilité que l'élément donné appartienne à un label.

Autrement dit, la fonction sigmoïde est la fonction de répartition de la régression logistique.

Soit $Y \in \{0, 1\}$ les différents labels que peut prendre l'élément que l'on considère et soit X l'ensemble des caractéristiques connues de l'élément, dont on cherche à déterminer dans quelle classe le mettre, donc quel label on doit lui attribuer. Soit θ le vecteur des poids des covariables, indiquant à quel point les covariables influencent sur la décision du label. On a donc:

$$P(Y = 1|X) = \frac{1}{1 + e^{-(X_1 \times w_1 + X_2 \times w_2 + \cdots + b)}}$$

et

$$P(Y = 0|X) = 1 - \frac{1}{1 + e^{-(X_1 \times w_1 + \cdots + b)}}$$

Pour plus de simplicité, on va considérer que le biais est compris dans les poids: au lieu d'écrire $z = w^T X + b$, on écrit $z = \theta^T \hat{X}$ avec $\hat{X} = \begin{bmatrix} 1 \\ X \end{bmatrix}$ modifié ou on a ajouté un 1 au début et $\theta = \begin{bmatrix} b \\ w \end{bmatrix}$. Ainsi, on a:

$$\theta^T \hat{X} = \theta_0 + X_1 \times \theta_1 + \cdots + X_n \times \theta_n = b + X_1 \times w_1 + \cdots + X_n \times w_n$$

Notre régression logistique binaire peut donc s'écrire comme:

$$P(Y = 1|X) = \frac{1}{1 + e^{\theta^T X}} = \sigma(\theta^T X)$$

et

$$P(Y = 0|X) = 1 - \sigma(\theta^T X)$$

1.0.0.1 Rapport de vraissemblance

Le rapport de vraissemblance est plus communément appelé odds ratio.

Si on veut essayer de définir le rapport de vraissemblance formellement, cela nous donne:

Soit $P(Y = 1|X) = p$ la probabilité que l'élément (possédant ses caractéristiques X) respecte une contrainte Y. On a donc $1-p$ la probabilité que l'élément ne respecte pas la contrainte Y. Le rapport de vraissemblance se définit de la sorte:

$$O(Y = 1|X) = \frac{p}{1-p}$$

On remarque:

$$\begin{array}{ll} \text{Si } p = 1-p \Rightarrow & O(Y = 1|X) = O(Y = 0|X) = 1 \\ \text{Si } p > 1-p \Rightarrow & O(Y = 1|X) > 1 \\ \text{Si } p < 1-p \Rightarrow & 0 \leq O(Y = 1|X) < 1 \text{ (car p, étant une probabilité, } \in [0, 1]) \end{array}$$

Ce qui signifie que si l'élément respecte bien la contrainte Y, alors le rapport de vraissemblance est supérieur à 1 et si l'élément ne respecte pas bien la contrainte Y, alors le rapport de vraissemblance est inférieur à 1.

Pour notre régression logistique binaire, le rapport de vraissemblance peut alors s'écrire:

$$\begin{aligned} & O(Y = 1|X) \\ &= \frac{P(Y = 1|X)}{P(Y = 0|X)} \\ &= \frac{\sigma(\theta^T X)}{1 - \sigma(\theta^T X)} \\ &= \frac{1}{1 + e^{-\theta^T X}} \times \frac{1}{1 - \frac{1}{1+e^{-\theta^T X}}} \\ &= \frac{1}{1 + e^{-\theta^T X}} \times \frac{1}{\frac{1+e^{-\theta^T X}-1}{1+e^{-\theta^T X}}} \\ &= \frac{1}{1 + e^{-\theta^T X}} \times \frac{1 + e^{-\theta^T X}}{e^{-\theta^T X}} \\ &= \frac{1}{e^{-\theta^T X}} \\ &= \boxed{e^{\theta^T X}} \end{aligned}$$

Le rapport de vraisemblance nous donne alors une indication sur le respect du label par notre élément.

1.1 Généralisation

On désire donc trouver une nouvelle distribution $\phi(z)$ tel que:

$$\phi(z) \in [0, 1] \ \forall z$$

est une généralisation de la fonction $\sigma(z)$

On veut donc que pour une régression logistique binaire, on ait $\sigma(z) = \phi(z)$.

On peut remarquer que:

$$P(Y = 1|X)$$

$$= \frac{1}{1 + e^{-\theta^T X}}$$
$$= \frac{1}{1 + e^{-\theta^T X}} \times \frac{e^{\theta^T X}}{e^{\theta^T X}}$$
$$= \frac{e^{\theta^T X}}{e^{\theta^T X} + e^{\theta^T X - \theta^T X}}$$
$$= \frac{e^{\theta^T X}}{e^{\theta^T X} + e^0}$$
$$= \frac{e^{\theta^T X}}{e^{\theta^T X} + 1}$$

On peut considérer que nous avons un vecteur de poids pour chaque label.

Ainsi, on a θ_0 pour le label 0 et θ_1 pour le label 1.

Comme on a besoin seulement d'un vecteur de poids pour déterminer le label de nouveaux éléments avec leurs caractéristiques, on peut considérer que $\theta_0 = 0$.

Ainsi, la formule précédente nous donne:

$$P(Y = 1|X)$$
$$= \frac{e^{\theta_1^T X}}{e^{\theta_1^T X} + 1}$$
$$= \frac{e^{\theta_1^T X}}{e^{\theta_1^T X} + e^0}$$
$$= \frac{e^{\theta_1^T X}}{e^{\theta_1^T X} + e^{0*X}}$$
$$= \frac{e^{\theta_1^T X}}{e^{\theta_1^T X} + e^{\theta_0^T X}}$$
$$= \frac{e^{\theta_1^T X}}{\sum_{i=0}^{1} e^{\theta_i^T X}}$$

On peut donc généraliser cette formule pour K labels.

Cela nous donne:

$$P(Y = k|X) = \frac{e^{\theta_k^T X}}{\sum_{i=0}^{K} e^{\theta_i^T X}}$$

Comme la fonction exponentielle est toujours positive, on a bien que:

$$0 \leq e^{\theta_k^T X} \leq e^{\theta_k^T X} + \sum_{i \neq k}^{K} e^{\theta_i^T X}$$

$$\Leftrightarrow 0 \leq e^{\theta_k^T X} \leq \sum_{i}^{K} e^{\theta_i^T X}$$

$$\Leftrightarrow 0 \leq \frac{e^{\theta_k^T X}}{\sum_{i}^{K} e^{\theta_i^T X}} \leq 1$$

$$\Leftrightarrow 0 \leq \phi(z) \leq 1$$

De plus, on a que:

$$\begin{aligned}
&\sum_{k}^{K} P(Y=k|X) \\
&= \sum_{k}^{K} \frac{e^{\theta_k^T X}}{\sum_i^K e^{\theta_i^T X}} \\
&= \frac{\sum_k^K e^{\theta_k^T X}}{\sum_i^K e^{\theta_i^T X}} \\
&= \frac{\sum_i^K e^{\theta_i^T X}}{\sum_i^K e^{\theta_i^T X}} \\
&= 1
\end{aligned}$$

Donc la fonction $\phi(z)$ est bien une fonction de distribution de probabilité qui généralise la fonction sigmoïde pour des problèmes à plusieurs labels.

Cette fonction est courramment appelée fonction `softmax`.

Notre objectif est donc de trouver une fonction de coût pour pouvoir entraîner les paramètres de la régression multinomiale. On cherche à maximiser la vraisemblance des données. Donc pour un label Y donné, on veut maximiser:

$$\sum_{k}^{K} f(Y,k) P(Y=k|X)$$

avec $f(Y,k)$ la fonction qui vaut 1 si $Y=k$ et 0 sinon.

En effet, en maximisant cette fonction, on fait en sorte que le paramètre θ_k permette d'obtenir la prédiction que le label soit égal à k avec la probabilité la plus grande possible. Afin de pouvoir utiliser un algorithme comme la descente en gradient, il faut non pas maximiser une fonction, mais minimiser une fonction.

C'est pourquoi, on peut utiliser l'inverse de cette fonction, dont on prend le logarithme pour simplifier les calculs, car on travaille avec des exponentielles.

Cela nous donne une fonction de coût comme suit:

$$\begin{aligned}
&\sum_{k}^{K} f(Y,k) log(\frac{1}{P(Y=k|X)}) \\
&\sum_{k}^{K} f(Y,k)(log(1) - log(P(Y=k|X))) \\
&\sum_{k}^{K} f(Y,k) - log(P(Y=k|X)) \\
&-\sum_{k}^{K} f(Y,k) log(P(Y=k|X))
\end{aligned}$$

On peut minimiser cette fonction de coût grâce à une descente en gradient.

Pour n données d'apprentissage, notre minimisation devient:

$$-\sum_{i}^{n}\sum_{k}^{K} f(Y_i,k) log(P(Y_i=k|X))$$