La régression logistique regularisée (à noyau)

Arthur Tenenhaus

12 octobre 2020

1 Introduction

Une population est divisée en 2 classes au moyen d'un critère qualitatif Y. Chaque individu de la population est décrit par p variables $X = (X_1, \dots X_p)$. La régression logistique est une méthode statistique adaptée à l'étude de la liaison entre la variable qualitative Y les p variables explicatives $X_1, X_2, \dots, X_p$. Ici, nous nous intéressons à la régression logistique régularisée à noyau.

Les notations suivantes sont utilisées :

— $(\mathbf{x}_1^\top, y_1), ..., (\mathbf{x}_n^\top, y_n)$, n réalisations indépendantes du couple (X, Y).
— $\mathbf{X} = (\mathbf{x}_1^\top, \dots, \mathbf{x}_n^\top)^\top$, la matrice des observations.
— $\mathbf{y} = (y_1, \dots, y_n)^\top$, le vecteur colonne de labels associés à chaque $\mathbf{x}_i$.

2 La régression logistique binaire multiple

2.1 Motivation

En préambule de la régression logistique, il convient de faire un bref rappel du modèle de régression linéaire. On cherche à expliquer Y par p variables explicatives $X = (1, X_1, ..., X_p)^\top$. Le modèle linéaire classique s'écrit :

$$Y = X^\top \boldsymbol{\beta} + \varepsilon = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p + \varepsilon$$

avec $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_p)^\top \in \mathbb{R}^{p+1}$ et $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ et on en déduit que

$$Y|X \sim \mathcal{N}(X^\top \boldsymbol{\beta}, \sigma^2)$$

Placons nous maintenant dans le cadre où la variable à prédire est qualitative (e.g. sexe, couleur, présence/absence d'une maladie). Cette variable possède un nombre fini de modalités. Le problème consiste alors à expliquer l'appartenance d'un individu à un groupe à partir des p variables explicatives. Il est bien entendu délicat de modéliser directement une variable Y (e.g. imaginons que Y soit le sexe d'une personne) par une relation linéaire et on s'intéressera plutôt aux probabilités d'appartenances des individus aux différentes classes. Dans cette séance nous nous intéressons au cas où la variable à expliquer prend uniquement

deux modalités (0 ou 1). Posons $\boldsymbol{x} = (1, x_1, ..., x_p)^\top$ le vecteur colonne formé de mesures des p variables explicatives X. La connaissance de $\mathbb{P}(Y = 1|X = \boldsymbol{x})$ implique donc celle de $\mathbb{P}(Y = 0|X = \boldsymbol{x})$. On cherche donc à modéliser $\pi(\boldsymbol{x}) = \mathbb{P}(Y = 1|X = \boldsymbol{x})$. On pourrait alors envisager de modéliser cette probabilité par une relation de la forme :

$$\pi(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

mais cette aproche possède quelques inconvénients dont le plus évident est que si aucune restriction n'est imposée au modèle, $\pi(\boldsymbol{x})$ peut prendre n'importe quelle valeur sur $\mathbb{R}$; ce qui peut être génant pour l'estimation d'une probabilité.... Fort de ce constat, en régression logistique binaire, on suppose que la probabilité $\pi(\boldsymbol{x})$ s'écrit sous la forme suivante :

$$\pi(\boldsymbol{x}) = \frac{\exp(\beta_0 + \beta_1 x_1 + ... + \beta_p x_p)}{1 + \exp(\beta_0 + \beta_1 x_1 + ... + \beta_p x_p)} \tag{1}$$

Soit en inversant la relation :

$$\text{logit}\left(\pi(\boldsymbol{x})\right) = \log\left(\frac{\pi(\boldsymbol{x})}{1 - \pi(\boldsymbol{x})}\right) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p \tag{2}$$

La probabilité d'observer la réponse y_i pour un individu ayant le vecteur $\boldsymbol{x_i} = (x_{i1}, ..., x_{ip})^\top$ est donc égale à :

$$\pi(\boldsymbol{x}_i) = \frac{\exp(\beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip})} \tag{3}$$

si la réponse y_i est égale à 1 et

$$1 - \pi(\boldsymbol{x}_i) = \frac{1}{1 + \exp(\beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip})} \tag{4}$$

si la réponse y_i est égale à 0.

Quelques commentaires sur le modèle logistique. Pour une observation $\boldsymbol{x}$ de la variable explicative X, on peut exprimer la variable d'intérêt comme suit :

$$Y = \pi(\boldsymbol{x}) + \varepsilon$$

Ici, la quantité ε peut prendre simplement deux valeurs : si $Y = 1$ alors $\varepsilon = 1 - \pi(\boldsymbol{x})$ avec probabilité $\pi(\boldsymbol{x})$ et $-\pi(\boldsymbol{x})$ avec probabilité $1 - \pi(\boldsymbol{x})$: $\mathbb{E}(\varepsilon) = 0$ et $\text{var}(\varepsilon) = \pi(\boldsymbol{x})(1 - \pi(\boldsymbol{x}))$; ce qui implique que la variance n'est pas constante et varie selon la valeur de $\boldsymbol{x}$ de X.
Pour définir le modèle logistique, nous effectuons deux choix :

1. Le choix d'une loi de Bernoulli pour $Y|X = \boldsymbol{x}$
2. Le choix de la modélisation de $\mathbb{P}(Y = 1|X = \boldsymbol{x})$ par

$$\text{logit}\left(\mathbb{P}(Y = 1|X = \boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{\beta}\right.$$

Question 1. Écrire la vraisemblance $L(\boldsymbol{\beta})$ du modèle.

Réponse à la question 1. Nous allons utiliser l'échantillon $(\boldsymbol{x}_1^\top, y_1), ..., (\boldsymbol{x}_n^\top, y_n)$ pour estimer le vecteur de paramètre $\boldsymbol{\beta}$ par la méthode du maximum de vraisemblance. Cette méthode consiste à chercher $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_p)^\top$ qui maximise la vraisemblance définie par :

$$\mathcal{L}(\beta) = \prod_{i=1}^{n} \mathbb{P}(Y = y_i | X = \boldsymbol{x}_i) = \prod_{i=1}^{n} \pi(\boldsymbol{x}_i)^{y_i} [1 - \pi(\boldsymbol{x}_i)]^{1-y_i} \tag{5}$$

Question 2. Montrer que l'estimateur du maximum de vraisemblance peut s'obtenir en considérant l'algorithme itératif suivant :

$$\boldsymbol{\beta}^{(s+1)} = \boldsymbol{\beta}^{(s)} + \left(\mathbf{X}^\top \mathbf{V} \mathbf{X}\right)^{-1} \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\pi}) \tag{6}$$

où $\boldsymbol{\pi}$ est le vecteur formé des $\pi_i = \pi(\mathbf{x}_i)$ estimé à l'itération courante, $\mathbf{X}$ la matrice formée d'une première colonne de coordonnées constantes égales à 1 et des p colonnes correspondant aux variables X_1, ..., X_p observées sur les n individus et $\mathbf{V}$ la matrice diagonale formée des $\pi_i(1 - \pi_i)$. Notons que ces équations sont résolues récursivement puisque que $\boldsymbol{\pi}$ et $\mathbf{V}$ évoluent à chaque itération.

Réponse à la question 2. On recherche $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_p)^\top$ maximisant la log-vraisemblance :

$$\begin{aligned} L(\boldsymbol{\beta}) &= \log(\mathcal{L}(\boldsymbol{\beta})) = \textstyle\sum_{i=1}^{n} \{y_i \log(\pi(\boldsymbol{x}_i)) + (1 - y_i) \log(1 - \pi(\boldsymbol{x}_i))\} \\ &= \sum_{i=1}^{n} \left\{ y_i \log\left(\frac{\pi(\boldsymbol{x}_i))}{1 - \pi(\boldsymbol{x}_i)}\right) + \log(1 - \pi(\boldsymbol{x}_i)) \right\} \\ &= \sum_{i=1}^{n} \left\{ y_i \underbrace{\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}\right)}_{\boldsymbol{x}_i^\top \boldsymbol{\beta}} - \log\left(1 + \exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}\right)\right) \right\} \end{aligned} \tag{7}$$

En annulant les dérivées de la log-vraisemblance par rapport au β_j , on aboutit au système d'équations suivant (appelé équation du score) :

$$\mathbf{U}(\boldsymbol{\beta}) = \begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \vdots \\ \frac{\partial L}{\partial \beta_p} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \iff \begin{cases} \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_0} = \sum_{i=1}^{n} (y_i - \pi_i) = 0 \\ \\ \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} x_{ij} (y_i - \pi_i) = 0, \quad j = 1, ..., p \end{cases} \tag{8}$$

qui n'a pas de solution analytique. En notant $\boldsymbol{\pi}$ le vecteur de probabilités tel que le ième élément égal π_i, on peut écrire les $p+1$ equations du score sous forme matricielle.

$$\mathbf{U}(\boldsymbol{\beta}) = \frac{\partial L}{\partial \boldsymbol{\beta}} = \mathbf{X}^\top(\mathbf{y} - \boldsymbol{\pi}) \tag{9}$$

La recherche des $\hat{\boldsymbol{\beta}}$ maximisant la log-vraisemblance s'effectue usuellement via l'algorithme de Newton-Raphson qui requiert le calcul des dérivées secondes de la log-vraisemblance. Notons $\mathbf{H}$, la matrice hessienne des dérivées secondes de la log-vraisemblance dont le terme général est défini par :

$$[\mathbf{H}(\boldsymbol{\beta})]_{jk} = \frac{\partial^2 L}{\partial \beta_j \partial \beta_k} = -\sum_{i=1}^{n} x_{ij} x_{ik} \frac{\exp(\boldsymbol{\beta}^\top \boldsymbol{x}_i)}{(1+\exp(\boldsymbol{\beta}^\top \boldsymbol{x}_i))^2} = -\sum_{i=1}^{n} x_{ij} x_{ik} \pi_i (1-\pi_i) \tag{10}$$

Par conséquent,

$$\mathbf{H}(\boldsymbol{\beta}) = \frac{\partial^2 L}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^\top} = -\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^\top \pi_i (1-\pi_i) = -\mathbf{X}^\top \mathbf{V} \mathbf{X} \tag{11}$$

On part tout d'abord d'une valeur initale arbitraire $\boldsymbol{\beta}^{(0)}$. Décrivons l'étape s. Considérons le développement de Taylor à l'ordre 2 de $L(\boldsymbol{\beta})$ autour de $\boldsymbol{\beta}^{(s)}$.

$$L(\boldsymbol{\beta}) \approx L(\boldsymbol{\beta}^{(s)}) + [\mathbf{U}(\boldsymbol{\beta}^{(s)})]^\top (\boldsymbol{\beta} - \boldsymbol{\beta}^{(s)}) + \frac{1}{2} (\boldsymbol{\beta} - \boldsymbol{\beta}^{(s)})^\top \mathbf{H}(\boldsymbol{\beta}^{(s)}) (\boldsymbol{\beta} - \boldsymbol{\beta}^{(s)}) \tag{12}$$

On choisit pour $\boldsymbol{\beta}^{(s+1)}$ la valeur de $\boldsymbol{\beta}$ maximisant le terme de droite de (12), obtenue en annulant sa dérivée :

$$\mathbf{U}(\boldsymbol{\beta}^{(s)}) + \mathbf{H}(\boldsymbol{\beta}^{(s)})(\boldsymbol{\beta}^{(s+1)} - \boldsymbol{\beta}^{(s)}) = 0 \tag{13}$$

D'où la construction de $\boldsymbol{\beta}^{(s+1)}$

$$\boldsymbol{\beta}^{(s+1)} = \boldsymbol{\beta}^{(s)} - [\mathbf{H}(\boldsymbol{\beta}^{(s)})]^{-1} \mathbf{U}(\boldsymbol{\beta}^{(s)}) \tag{14}$$

L'algorithme de Newton Raphson converge vers une estimation $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_p)$ maximisant la log-vraisemblance.

En utilisant les équations (9) et (11), l'étape courante de l'algorithme de Newton-Raphson peut s'écrire comme suit :

$$\begin{aligned}
\boldsymbol{\beta}^{(s+1)} &= \boldsymbol{\beta}^{(s)} + \left(\mathbf{X}^\top \mathbf{V} \mathbf{X}\right)^{-1} \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\pi}) \\
&= \left(\mathbf{X}^\top \mathbf{V} \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{V} \left(\mathbf{X}\boldsymbol{\beta}^{(s)} + \mathbf{V}^{-1}(\mathbf{y} - \boldsymbol{\pi})\right) \\
&= \left(\mathbf{X}^\top \mathbf{V} \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{V} \mathbf{z}
\end{aligned}$$

Les lignes 2 et 3 précédentes ont pour objectif de formuler l'étape de Newton comme une méthode de régression pondérée, avec pour réponse

$$\mathbf{z} = \mathbf{X}\boldsymbol{\beta}^{(s)} + \mathbf{V}^{-1}(\mathbf{y} - \boldsymbol{\pi}) \tag{15}$$

Dans le cas précis de la régression logistique l'algorithme de Newton-Raphson est appelé IRLS (Iterative Reweighted Least Squares). Ces équations sont résolues récursivement puisque que $\boldsymbol{\pi}$ évolue à chaque itération et ainsi $\mathbf{V}$ et $\mathbf{z}$. La fonction `myLR()` présentée ci-dessous implémente l'algorithme de Newton-Raphson en suivant le formalisme IRLS.

```
myLR = function(X, y, tolerance = 1e-6, max.iter=200){
  X = cbind(1, X)
  beta_s = rep(0, NCOL(X))
  pi = runif(NROW(X), 0, 1)
  V = diag(pi*(1-pi))
  iter = 1

  made.changes = TRUE

  while (made.changes & (iter < max.iter))
  {
    iter = iter + 1
    made.changes <- FALSE
    beta_s_plus_1 = beta_s + solve(t(X)%*%V%*%X)%*%t(X)%*%(y-pi)

    pi = drop(1/(1+exp(-X%*%beta_s_plus_1)))
    V = diag(pi*(1-pi))

    relative.change = drop(crossprod(beta_s_plus_1 - beta_s))
     /drop(crossprod(beta_s))
    made.changes = (relative.change > tolerance)

    beta_s = beta_s_plus_1

    if (iter == 200)
      warning("The Newton-Raphson algorithm did not converge
          after 200 iterations.")

  }
  if (iter< 200)
    print(paste("The Newton-Raphson algorithm converges after",
                iter, "iterations"))

  return(list(beta = beta_s , proba = pi))
}
```

Remarque : La fonction **glm()** disponible dans le package **stats** implémente le modèle linéaire généralisé. La régression logistique binaire comme cas particulier du modèle linéaire généralisé est donc disponible via cette fonction.

Question 3. Justifier que l'algorithme converge vers l'unique estimateur du maximum du vraisemblance.

Réponse à la question 3. La matrice des dérivées secondes de la log-vraisemblance étant définie négative, on peut en conclure que la log-vraisemblance est une fonction concave et possède donc un maximum en annulant le vecteur de score $\mathbf{U}(\beta)$ formée des dérivées premières.
La méthode de Newton-Raphson permet une résolution numérique des équations du score et permet de construire une suite $\beta^{(s)}$ convergeant vers l'estimation du maximum de vraisemblance. La concavité à une conséquence numérique importante puisqu'elle justifie que l'algorithme de Newton-Raphson converge vers un maximum global de la vraisemblance. De plus, la convergence de l'algorithme ne dépend pas du point initial.

On constate que cette algorithme requiert l'inversion d'une matrice $p \times p$, différente à chaque itération de surcroît. Pour contourner cette étape calculatoire délicate en grande dimension, on se propose d'approximer la matrice $\mathbf{H}_1 = -\mathbf{X}^\top \mathbf{V}\mathbf{X}$ par une matrice $\mathbf{H}_2$ ne dépendant pas de l'itération. Pour garantir la convergence de l'algorithme, on peut montrer qu'il suffit que la matrice $\mathbf{H}_1 - \mathbf{H}_2$ soit définie positive.

Question 4. En remarquant que $\pi_i(1-\pi_i)$ est majoré par $\frac{1}{4}$, proposer une approximation $\mathbf{H}_2$ de la matrice $\mathbf{H}_1$ telle que $\mathbf{H}_1 - \mathbf{H}_2$ soit définie positive.

Question 5. Réécrire l'algorithme itératif décrit par l'équation (6) en y injectant cette approximation.

On considère maintenant la vraisemblance pénalisée définie comme suit :

$$L_\lambda(\boldsymbol{\beta}) = L(\boldsymbol{\beta}) - \frac{\lambda}{2} \|\boldsymbol{\beta}_\lambda\|_2^2 \tag{16}$$

où le paramètre λ est un paramètre de régularisation.

Question 6. Discuter l'intérêt de considérer la maximisation de la vraisemblance pénalisée.

Question 7. Montrer que la maximisation de la vraisemblance pénalisée est revient à résoudre le problème problème d'optimisation suivant :

$$\min_\beta \sum_{i=1}^{n} \ln(1 + \exp(-y_i x_i^\top \boldsymbol{\beta})) + \frac{\lambda}{2} \|\boldsymbol{\beta}\|_2^2 \tag{17}$$

Indication : Encoder la classe d'appartenance en $y_i = \{-1, 1\}$ (plutôt qu'en $\{0, 1\}$).

Question 8. Montrer que la maximisation de la vraisemblance pénalisée peut s'obtenir en considérant l'algorithme itératif suivant :

$$\boldsymbol{\beta}_\lambda^{(s+1)} = \boldsymbol{\beta}_\lambda^{(s)} + 4\left(\mathbf{X}^\top\mathbf{X} + 4\lambda\mathbf{I}_p\right)^{-1}\left(\mathbf{X}^\top(\mathbf{y}-\boldsymbol{\pi}) - \lambda\boldsymbol{\beta}_\lambda^{(s)}\right) \tag{18}$$

On constate toutefois, que l'équation (18) nécessite l'inversion d'une matrice de taille $p \times p$, ce qui peut-être limitant en grande dimension. Pour contourner les problèmes de calculs liés à la grande dimension, nous allons proposer une écriture duale de $\boldsymbol{\beta}_\lambda^{(s)}$.

Question 9. En supposant que $\mathbf{X}\mathbf{X}^\top$ est de rang plein, montrer qu'une version duale de l'algorithme de régression logistique régularisée est définie par :

$$\boldsymbol{\alpha}_\lambda^{(s+1)} = \boldsymbol{\alpha}_\lambda^{(s)} + 4\left(\mathbf{X}\mathbf{X}^\top + 4\lambda\mathbf{I}_n\right)^{-1}\left(\mathbf{y} - \boldsymbol{\pi} - \lambda\boldsymbol{\alpha}_\lambda^{(s)}\right) \tag{19}$$

3 Cas pratique

L'exemple compagnon de ce TP est le jeu de donnés réel "Alzheimer". Vous trouverez un descriptif détaillé de ce jeu de données dans [Webster et al., 2009]. Vous trouverez le jeu de données et le papier associé sur Edunao.
Il s'agit d'un jeu de données d'expression de gènes mesurées sur 188 contrôles versus 176 patients atteint de la madadie d'Alzheimer. On souhaite prédire le statut du patient à partir de l'expression de 8650 gènes.

Question 10. Implémenter à l'aide du logiciel de votre choix, la version de la régression logistique régularisée qui vous paraît la plus adaptée aux dimensions du jeu de données "Alzheimer".

Question 11. Dans une boucle de validation croisée, entraîner une régression logistique régularisée. Tracer l'évolution du taux d'erreur de classification en apprentissage et en test en fonction des paramètres des hyper-paramètres du modèle.

Références

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. H. (2001). *The elements of statistical learning : data mining, inference, and prediction.* New York : Springer-Verlag.

[Webster et al., 2009] Webster, J. A., Gibbs, J. R., Clarke, J., Ray, M., Zhang, W., Holmans, P., Rohrer, K., Zhao, A., Marlowe, L., Kaleem, M., et al. (2009). Genetic control of human brain transcript expression in alzheimer disease. *The American Journal of Human Genetics*, 84(4) :445–458.