Université Grenoble Alpes

Master SSD, BDA et IC
Enseignant : Jonathan El Methni
Apprentissage Statistique

TP 6 : Modèles linéaires pénalisés et glmnet

L'objectif de ce TP est de se familiariser avec la régression pénalisée en utilisant le package "glmnet" mais aussi en illustrant le recours à ce type de traitement pour résoudre un problème récurrent pour lequel d'autres méthodes "classiques" ne donnent pas de solutions satisfaisantes.

L'étude portera sur le fichier "Autos" que l'on peut importer via la fonction Import Dataset de RStudio (en haut à droite) afin de lire le fichier Autos.txt. Cliquer ensuite sur From Text (base). Une fois le fichier sélectionné cliquer sur Row names et sélectionner Use first column.

La variable à expliquer est Y (PRIX). On définit la matrice X comme étant la matrice des prédicteurs c'est-à-dire les variables explicatives.

Préambule

1. Donner un résumé succinct des données. Indiquer le nombre n des individus et le nombre de variables.
2. Bien que Ridge et Lasso traitent aussi bien les variables qualitatives que les variables quantitatives on ne considérera (dans la liste des prédicteurs) que les variables quantitatives. Nettoyer votre fichier de données en éliminant les variables qualitatives.

```
Autos = Autos[, -c(7,8)]
```

Regression linéaire multiple

1. Effectuer une régression linéaire multiple par moindres carrés ayant pour but d'expliquer le prix (variable PRIX) par les autres caractéristiques des voitures.
   (a) Est-ce qu'un test global de Fisher indique que le modèle explique le PRIX de manière significative mieux que le modèle nul (sans variables explicatives) ?
   (b) Y a-t-il des variables significatives ?
   (c) Quel est le problème sous jacent ? Explicitez le.
   (d) Calculer le VIF en utilisant le package "car" et commentez.
   (e) Confirmer l'existence de ce problème à l'aide d'une ACP.
2. Quel est la conséquence importante de ce problème en apprentissage ?

Préambule aux méthodes pénalisées

1. Extraire la matrice X des prédicteurs du fichier de données à l'aide de l'instruction suivante

```
X = model.matrix(PRIX~., Autos)[,-1]
```

2. Standardiser X et installer/charger le package "glmnet".

Régression ridge

1. Effectuer une régression ridge ayant pour but d'expliquer le prix par les autres caractéristiques des voitures.
2. Donner le chemin de régularisation en fonction de la norme L_2.
3. Donner le chemin de régularisation en fonction de $\log(\lambda)$.
4. Pour les deux précédents graphiques :
   (a) que représentent les nombres sur l'axe vertical à gauche ? Comment les obtient-on ?
   (b) que représentent les nombres en haut du graphe ? Pourquoi sont-ils égaux ?
5. Afficher la grille des λ générée par défaut par glmnet pour calculer les estimations des coefficients.
6. Afficher les valeurs des coefficients estimés par glmnet pour la 5eme et la 58eme valeur de λ.

Régression lasso

1. Effectuer une régression lasso ayant pour but d'expliquer le prix par les autres caractéristiques des voitures.
2. Donner le chemin de régularisation en fonction de la norme L_1.
3. Donner le chemin de régularisation en fonction de $\log(\lambda)$.
4. Pour les deux précédents graphiques que représentent les nombres en haut du graphe ?
5. Afficher la grille des λ générée par défaut par glmnet pour calculer les estimations des coefficients.
6. Afficher les valeurs des coefficients estimés par glmnet pour la 5eme et la 58eme valeur de λ.
7. Tracer le nombre de variables sélectionnées en fonction de λ.

Régression elastic-net

1. Effectuer une régression Elastic-net avec un $\alpha = 0.3$ ayant pour but d'expliquer le prix par les autres caractéristiques des voitures.
2. Donner le chemin de régularisation en fonction de la norme L_1.
3. Donner le chemin de régularisation en fonction de $\log(\lambda)$.
4. Pour les deux précédents graphiques que représentent les nombres en haut du graphe ?
5. Afficher la grille générée des λ par défaut par glmnet pour calculer les estimations des coefficients.
6. Afficher les valeurs des coefficients estimés par glmnet pour la 5eme et la 58eme valeur de λ.
7. Refaire la même chose avec un $\alpha = 0.7$
8. Pour les deux valeurs de α tracer le nombre de variables sélectionnées en fonction de λ et comparer les résultats.

Selection du paramètre λ

La fonction cv.glmnet() permet de sélectionner λ par validation croisée. Afin d'avoir toujours les mêmes folds de validation on réalise un découpement fixer à l'aide de l'instruction suivante :

```
foldidT=sample(rep(seq(6),length=n))
```

Dans la suite, à chaque utilisaiton de cv.glmnet on précisera foldid=foldidT.

1. Sélection du paramètre λ pour ridge :
   (a) Réaliser une validation croisée à l'aide de cv.glmnet pour une régression ridge.
   (b) Tracer le graphe des erreurs quadratiques de prévision (MSE) en fonction de $\log(\lambda)$.
   (c) A quoi correspondent les deux droites verticales représentées en pointillés ? Donner les valeurs des λ et de la MSE correspondants.
   (d) Faire la régression ridge correspondant au λ optimal.
   (e) Afficher les coefficients ainsi calculés. A quoi correspond "Intercept" ?

2. Sélection du paramètre λ pour lasso :
   (a) Réaliser une validation croisée à l'aide de cv.glmnet pour une régression lasso.
   (b) Tracer le graphe des erreurs quadratiques de prévision en fonction de $\log(\lambda)$.
   (c) Faire la régression lasso correspondant au λ optimal.
   (d) Afficher les coefficients ainsi calculés. Interpréter.

3. On pourrait faire la même chose pour elastic-net à $\alpha = 0.7$ fixé.

Comparaison des performances

1. Afin de mieux interpréter les résultats, représenter sur une même figure l'évolution des performances de validation croisée obtenue par les deux modèles ridge et lasso quand le paramètre λ de régularisation varie. Quel modèle offre les meilleures performances ?

2. Reproduire cette analyse en considérant une penalité elastic-net afin de considérer un compromis entre ridge et lasso. Commenter les résultats. Il suffit pour cela de modifier le paramètre α de la fonction cv.glmnet() et de le faire varier entre 0 (ridge) et 1 (lasso). On pourra par exemple considérer une grille définie selon un pas de 0.2.

3. On veut dresser un bilan des résultats obtenus en fonction des meilleures performances de validation croisée et du nombre de variables sélectionnées.

   Pour chaque modèle on retiendra la valeur lambda.1se proposée par la fonction cv.glmnet() comme "meilleur" paramètre de régularisation. Notons que le champ nzero de l'objet renvoyé par cv.glmnet() donne le nombre de coefficients non-nuls pour chaque valeur de λ considérée (l'ensemble de ces valeurs étant stockées dans le champ lambda).

   (a) Tracer les diagrammes (en bâtons) du nombre de variables sélectionnées et de la performance en fonction de la séquence des valeurs de α.
   (b) Commenter les résultats.
   (c) Quelle valeur de α retiendrez- vous ?
   (d) Comparer le chemin de régularisation correspondant au choix de α à celui du lasso.

Prédictions

Pour un lambda choisi, la fonction cv.glmnet réalise un réajustement du modèle sur toutes les données.

Il suffit par conséquent d'appliquer la fonction predict à l'objet obtenu avec cv.glmnet en spécifiant la valeur de lambda souhaitée.

1. Réaliser les prédictions sur tout le tableau de données pour ridge, lasso et elastic-net ($\alpha = 0.7$), ceci à chaque fois pour le λ optimal. Commenter.

2. Enfin, éaliser une prédiction pour une nouvelle voiture dont les caractéristiques sont :
   - CYL = 1300
   - PUIS = 70
   - LON = 425
   - LAR = 169
   - POIDS = 1050
   - VITESSE = 150