4 jeux de données

Les indicateurs

- tracer le nuage de points et la droite ajustée pour examiner la linéarité.
- utiliser la fonction `acf` pour exmainer la baisse de l'autocorrélation des résidus.
- tracer les résidus en fonction de l'ordre de x .
- tracer un histogramme des résidus et vérifier qu'il est gaussien autour de 0 et un Q-Q plot marche aussi.

```
# creates a data frame named datum from imported csv file
datum=read.csv(file.choose())
# check data was imported properly
# column headers of data file plus first 6 rows of data
head(datum)
# summary statistics for datum
summary(datum)
# runs a linear regression, calls results 'results'
results=lm(datum[,2]~datum[,1],data=datum)
summary(results) # prints a summary of 'results'
```

Hypothèse non-vérifiée par jeu de données

```
+ data1.csv :  problème d'homoscédasticité.
+ data2.csv :  corrélation des erreurs.
+ data3.csv :  normalité des résidus.
+ data4.csv :  relation linéaire.
```

Sélection de modèle et prédiction

Dans cette partie, on s'intéresse au jeu de données du cancer de la prostate disponible dans le `package lasso2` de R. Nous avons besoin d'installer le package `lasso2` et charger le jeu de données comme suit

```
install.packages("lasso2", dep=TRUE)
require(lasso2)
data("Prostate")
?Prostate
```

Nous allons construire un modèle linéaire multiple pour prédire la variable `lcavol` en fonction des autres variables. Le jeu de données est constitué de 97 observations. Nous allons le diviser en deux parties : un jeu de données apprentissage constitué des 70 premières observations et un jeu de données test constitué des 27 dernières observations du tableau.

Sur le jeu de données apprentissage et à l'aide de la fonction `step` de R

1. Mettre en oeuvre trois procédures de choix de modèle, *forward*, *backward* et dans les deux sens (*forward* et *backward*) pour sélectionner le meilleur modèle au sens du critère AIC.

```
require(lasso2)
data("Prostate")
n <- nrow(Prostate)
Prostate.train <- Prostate[1:70,]
Prostate.test <- Prostate[71:97,]
#### AIC
```

```
## forward
lm0 <- lm(lcavol ~ 1, data = Prostate.train)
lm1 <- formula(lm(lcavol~ ., data = Prostate.train))
slm.for.aic <- step(lm0, direction = "forward", scope=lm1)
## backward
lm1 <- lm(lcavol ~ ., data = Prostate.train)
slm.back.aic <- step(lm1, direction = "backward")
##both
slm.both.aic <- step(lm1, direction = "both", scope = list(lm1, lm0))
```

2. Modifier les trois procédures précédentes pour sélectionner le meilleur modèle au sens du critère BIC.

```
#### BIC
## forward
lm0 <- lm(lcavol ~ 1, data = Prostate.train)
lm1 <- formula(lm(lcavol ~ ., data = Prostate.train))
slm.for.bic <- step(lm0, direction = "forward", scope=lm1, k=log(n))
## backward
lm1 <- lm(lcavol ~ ., data = Prostate.train)
slm.back.bic <- step(lm1, direction = "backward", k=log(n))

##both
slm.both.bic <- step(lm1, direction = "both", scope = list(lm1, lm0),k=log(n))
```

3. Quel est le modèle préférable ? Justifier.

réponse

On préfère le modèle correspondant au critère bic car plus parcimonieux.

Nous allons maintenant comparer les erreurs de prédiction des trois modèles : le modèle sélectionné par critère AIC, le modèle sélectionné par le critère BIC et le modèle complet. Pour cela nous allons faire appel à la fonction `predict` de R. Pour calculer une erreur de prédiction, il suffit d'utiliser la quantité

```
mean((x-y)**2)
```

où `x` et `y` sont deux vecteurs de tailles égales. On peut assimiler `y` au vecteur des vraies valeurs observées d'une certaine variables et `x` les prédictions de ces différentes valeurs.

Comparaison des erreurs de prédiction sur le jeu de données apprentissage

- Calculer cette erreur pour les trois modèles (AIC, BIC et modèle complet).

```
mean((predict(slm.both.bic)-Prostate.train[,"lcavol"])**2)
mean((predict(slm.both.aic)-Prostate.train[,"lcavol"])**2)
mean((predict(lm1)-Prostate.train[,"lcavol"])**2)
```

- Expliquer la hiérarchie observée ?

réponse

La hiérarchie observée s'explique essentiellement par le nombre de variables incluses dans le modèle.

Comparaison des erreurs de prédiction sur le jeu de données test

- Conserver les modèles sélectionnés avec le jeu de données apprentissage et calculer l'erreur de prédiction des trois modèles (AIC, BIC et modèle complet) sur le jeu de données test.

réponse

```
mean((predict(slm.both.bic, newdata = Prostate.test)-Prostate.test[,"lcavol"])**2)
mean((predict(slm.both.aic, newdata = Prostate.test)-Prostate.test[,"lcavol"])**2)
mean((predict(lm1, newdata = Prostate.test)-Prostate.test[,"lcavol"])**2)
```

- Quel modèle peut-on recommander pour une meilleure prédiction ?

réponse

On remarque que le modèle sélectionné par le critère BIC possède la meilleure de prédiction de plus, il fait appel à un nombre minimal de covariables.