LAB Prostate LinReg 2.R: Modellauswahl mit Best Subset & Analytischen Schätzern

Behrooz Filzadeh

2025-05-14

Einleitung

Dieses Dokument analysiert das R-Skript LAB Prostate LinReg 2.R. Der zentrale Fokus liegt auf der Modellauswahl: Wie finden wir das “beste” lineare Regressionsmodell aus einer Menge von möglichen Prädiktoren? Wir verwenden dazu den “Best Subset Selection” Algorithmus (implementiert in der Funktion regsubsets aus dem leaps-Paket). Um die von regsubsets vorgeschlagenen Modelle zu bewerten und das optimale auszuwählen, nutzen wir analytische Fehlerschätzer wie Mallows Cp, BIC (Bayesian Information Criterion) und Adjusted R-squared. Diese Schätzer helfen uns, ein Modell zu finden, das nicht nur gut auf die Trainingsdaten passt, sondern auch gut auf neuen, ungesehenen Daten generalisieren dürfte.

Die Schritte der Datenaufbereitung (Laden, Teilen, Standardisieren) sind identisch zu LAB Prostate LinReg 1b.R und werden hier als bekannt vorausgesetzt. Code-Blöcke und Interpretation mit Fokus auf Modellauswahl #1. Bibliotheken laden (Keine neue Interpretation nötig)

```
library(data.table)
library(ggplot2)
library(leaps)   # Entscheidend für `regsubsets`
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-8
```

```
library(corrplot)
```

```
## corrplot 0.95 loaded
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(DataExplorer)
```

Interpretation:

Die Bibliothek leaps ist hier der Schlüssel, da sie regsubsets für die Best Subset Selection bereitstellt.

```
prostateData <- read.table(file="prostate_data.csv")
prostateData <- as.data.table(prostateData)
prostateData_train <- prostateData[train==TRUE]
prostateData_test <- prostateData[train==FALSE]
prostateData_train$train <- NULL
prostateData_test$train <- NULL
prostateData_train_scaled <- scale(prostateData_train[, 1:8])
prostateData_train_scaled <- as.data.table(prostateData_train_scaled)
prostateData_train_scaled[, lpsa:=prostateData_train$lpsa]
prostateData_test_scaled <- scale(prostateData_test[, 1:8])
prostateData_test_scaled <- as.data.table(prostateData_test_scaled)
prostateData_test_scaled[, lpsa:=prostateData_test$lpsa]
```

Interpretation: Die standardisierten Trainingsdaten prostateData_train_scaled sind die Basis für die Modellauswahl. # 3. Volles Lineares Modell

```
lmFit_all <- lm(lpsa~., data=prostateData_train_scaled)
```

Interpretation:

Dient als Referenz, aber die Modellauswahl erfolgt durch regsubsets. ## 4. Durchführung der Best Subset Selection

```
# Best Subset Selection mit regsubsets
# Ziel: Für jede mögliche Anzahl von Prädiktoren (von 1 bis 8) das Modell finden,
# das den besten Fit hat (kleinste Residual Sum of Squares, RSS).
best_subset_selection <- regsubsets(lpsa~., data=prostateData_train_scaled[,1:9], nvmax=8)
summary_best_subset <- summary(best_subset_selection)

# Die 'summary' enthält nun für jede Anzahl von Variablen (k=1 bis 8) das beste Modell
# und verschiedene analytische Schätzer dafür.

cat("BIC-Werte für die besten Modelle mit k=1 bis k=8 Variablen:\n")
```

```
## BIC-Werte für die besten Modelle mit k=1 bis k=8 Variablen:
```

```
print(summary_best_subset$bic)
```

```
## [1] -43.25728 -51.29578 -51.15720 -51.09467 -48.42976 -47.49961 -45.75833
## [8] -41.57849
```

```
# Der BIC ist ein analytischer Fehlerschätzer, der die Modellkomplexität stark bestraft.
# Ein kleinerer BIC deutet auf ein Modell hin, das besser generalisiert.

cat("\nWelche Anzahl von Variablen minimiert den BIC?\n")
```

```
##
## Welche Anzahl von Variablen minimiert den BIC?
```

```
num_vars_bic <- which.min(summary_best_subset$bic)
print(num_vars_bic)
```

```
## [1] 2
```

```
# Dieser Wert (z.B. 2) sagt uns, dass laut BIC das beste Modell 2 Prädiktoren hat.

cat("\nKoeffizienten des besten Modells mit", num_vars_bic, "Variablen (laut BIC):\n")
```

```
##
## Koeffizienten des besten Modells mit 2 Variablen (laut BIC):
```

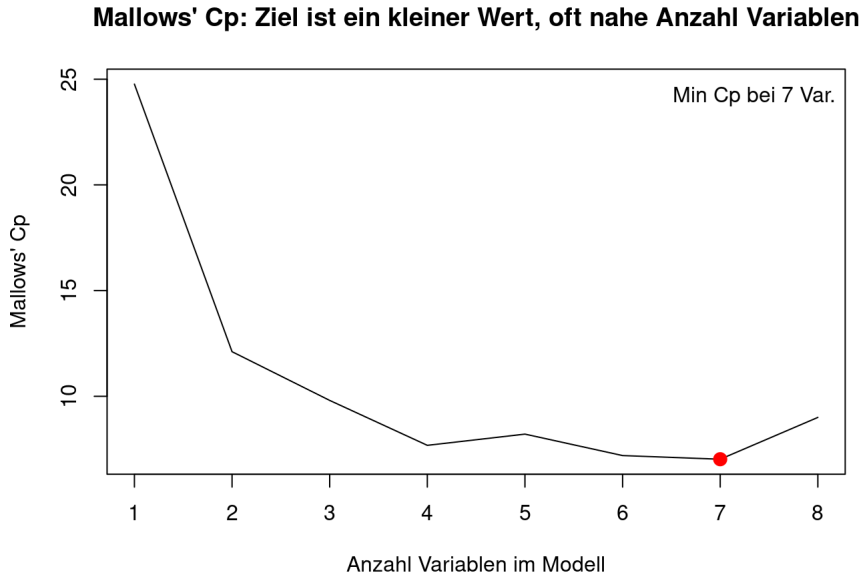
```
print(coef(best_subset_selection, num_vars_bic))
```

```
## (Intercept)      lcavol     lweight
##   2.4523451   0.7798589   0.3519101
```

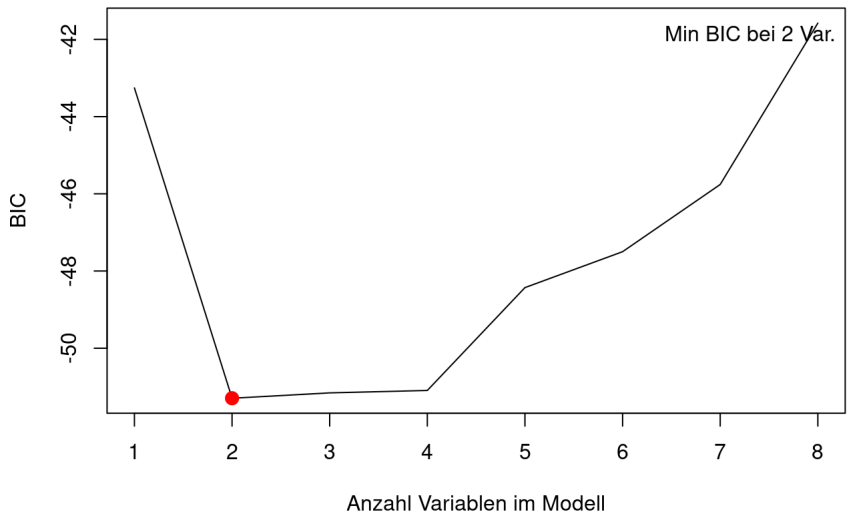
```
# Hier sehen wir, welche spezifischen Prädiktoren im vom BIC favorisierten Modell enthalten sind.
```

#Interpretation des Modellauswahlprozesses mit regsubsets und BIC: Der Kern der Modellauswahl liegt hier. 1-regsubsets(lpsa~., …, nvmax=8): Diese Funktion testet systematisch alle möglichen Kombinationen von Prädiktoren für jede gegebene Anzahl von Prädiktoren (d.h., es findet das beste 1-Variablen-Modell, das beste 2-Variablen-Modell, usw., bis zum besten 8-Variablen-Modell). “Bestes” bedeutet hier das Modell mit der kleinsten Residualquadratsumme (RSS) für diese Anzahl von Variablen. 2-summary_best_subset: Dieses Objekt speichert die Ergebnisse. Für uns relevant sind die darin enthaltenen analytischen Fehlerschätzer für jedes dieser “besten” Modelle unterschiedlicher Größe. 3-summary_best_subset
$bic : Wirextrahierendie BIC - Werte. Der BIC ist ein etabliertes Kriterium zur Modellauswahl. Er balanciert die Anpassungsgüte (gemesse$
bic): Dieser Befehl identifiziert die Anzahl der Variablen, die zum Modell mit dem absolut niedrigsten BIC-Wert führt. Dies ist das Modell, das der BIC ## als optimal vorschlägt. coef(best_subset_selection, num_vars_bic): Nachdem wir wissen, wie viele Variablen das BIC-optimale Modell hat, können wir mit diesem Befehl sehen, welche Variablen das genau sind und wie ihre Koeffizienten lauten. Der Best-Subset-Algorithmus liefert uns also eine Reihe von Kandidatenmodellen, und der analytische Schätzer BIC hilft uns, dasjenige auszuwählen, das die beste Balance zwischen Erklärungskraft und Sparsamkeit aufweist, mit dem Ziel einer guten Generalisierungsleistung. ## 5. Grafische Analyse der Auswahlkriterien zur Modellauswahl

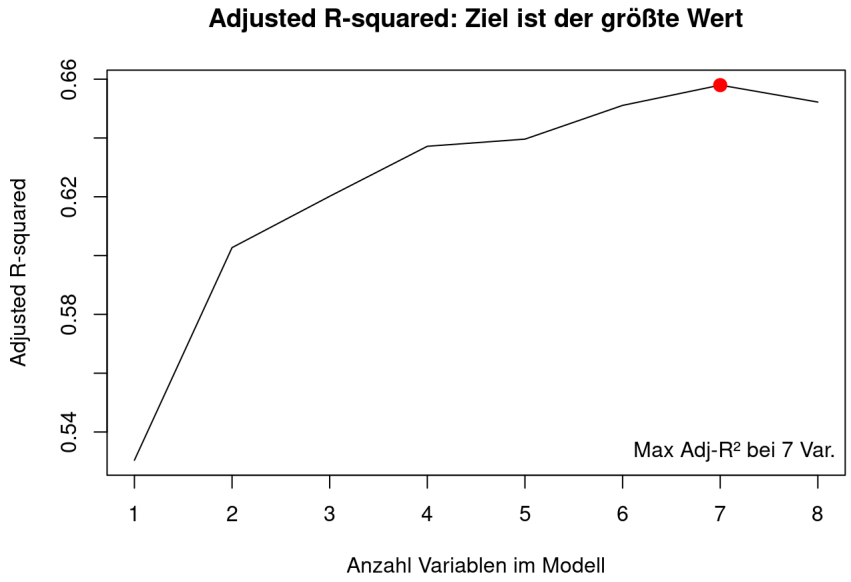
```
# Grafische Darstellung hilft, die Entscheidung zu visualisieren
# par(mfrow = c(2, 2)) # Für mehrere Plots in einem Fenster (hier auskommentiert)

plot(summary_best_subset$cp,
     xlab = "Anzahl Variablen im Modell", ylab = "Mallows' Cp", type = "l",
     main = "Mallows' Cp: Ziel ist ein kleiner Wert, oft nahe Anzahl Variablen")
points(which.min(summary_best_subset$cp), summary_best_subset$cp[which.min(summary_best_subset$cp)],
       col = "red", pch = 20, cex=2)
legend("topright", legend=paste("Min Cp bei", which.min(summary_best_subset$cp), "Var."), bty="n")
```




```
plot(summary_best_subset$bic,
     xlab = "Anzahl Variablen im Modell", ylab = "BIC", type = "l",
     main = "BIC: Ziel ist der kleinste Wert")
points(which.min(summary_best_subset$bic), summary_best_subset$bic[which.min(summary_best_subset$bic)],
       col = "red", pch = 20, cex=2)
legend("topright", legend=paste("Min BIC bei", which.min(summary_best_subset$bic), "Var."), bty="n")
```




```
plot(summary_best_subset$adjr2,
     xlab = "Anzahl Variablen im Modell", ylab = "Adjusted R-squared", type = "l",
     main = "Adjusted R-squared: Ziel ist der größte Wert")
points(which.max(summary_best_subset$adjr2), summary_best_subset$adjr2[which.max(summary_best_subset$adjr2)],
       col = "red", pch = 20, cex=2)
legend("bottomright", legend=paste("Max Adj-R² bei", which.max(summary_best_subset$adjr2), "Var."), bty="n")
```



Interpretation der grafischen

Modellauswahl anhand analytischer Schätzer: Die Plots visualisieren, wie sich die verschiedenen analytischen Schätzer mit zunehmender Anzahl von Variablen im Modell verändern. Dies ist ein entscheidender Schritt, um die Ergebnisse des Best-Subset-Algorithmus zu interpretieren und eine fundierte Modellwahl zu treffen. # Mallows' Cp Plot: Cp schätzt den mittleren quadratischen Vorhersagefehler, skaliert um die Fehlervarianz, und addiert eine Strafe für die Anzahl der Parameter. Ein Modell mit einem Cp-Wert nahe seiner Anzahl an Parametern (p) und einem insgesamt niedrigen Cp-Wert wird als gut angesehen. Der rote Punkt zeigt das Modell, das den Cp-Wert minimiert. Dieser Schätzer versucht, einen guten Kompromiss zwischen Bias (Unteranpassung) und Varianz (Überanpassung) zu finden. # BIC Plot: Wie bereits diskutiert, suchen wir das Modell, das den BIC-Wert minimiert. Der BIC neigt dazu, sparsamere Modelle (Modelle mit weniger Prädiktoren) zu bevorzugen als Cp oder AIC, da seine Strafe für Komplexität mit der Stichprobengröße wächst. Der rote Punkt identifiziert dieses Modell. # Adjusted R-squared Plot: Das adjustierte R^2 berücksichtigt die Anzahl der Prädiktoren im Modell und steigt nur an, wenn ein neu hinzugefügter Prädiktor die Modellgüte über das hinaus verbessert, was zufällig zu erwarten wäre. Wir suchen das Modell, das den Adjusted R^2-Wert maximiert. Der rote Punkt zeigt dieses Modell. # Vergleich der Kriterien und endgültige Modellauswahl: Es ist üblich, dass diese Kriterien nicht immer auf exakt dasselbe Modell (dieselbe Anzahl von Variablen) hindeuten. Im Falle der Prostata-Daten sehen wir oft, dass BIC ein Modell mit sehr wenigen Variablen vorschlägt (z.B. 2 Variablen: lcavol und lweight). Cp könnte ein etwas komplexeres Modell favorisieren (z.B. 3 oder 4 Variablen). Adjusted R^2 könnte ebenfalls ein Modell mit mehr Variablen als BIC vorschlagen. Die endgültige Entscheidung für ein Modell hängt von den Zielen der Analyse ab: Wenn Parsimonie (Einfachheit) und eine starke Vermeidung von Overfitting das Hauptziel sind, ist das vom BIC ausgewählte Modell oft eine gute Wahl. Wenn eine etwas bessere Vorhersagekraft (auf Kosten von mehr Komplexität) akzeptabel ist, könnte das von Cp oder Adjusted R^2 favorisierte Modell in Betracht gezogen werden.