Chapter 10: Regression

Wang Shujia

Contents

1 Regression Model and Estimation

1.1 Model

Workflow For Building a Regression Model

Regression analysis is used for modeling the relationship between a single variable Y , called the **response** or **dependent** variable, and one or more **explanatory** variables, also called **predictor**(s) or **independent** variable(s).

1. Describing the Data. Use the *plot()*, *boxplot()* and *summary()* commands
2. Data Transformation (If needed). Use the *log* command
3. Model. Use the *lm()* and *summary()* commands
4. Diagnostics. Fitted values vs standardised residuals. Influence and Cook's distance
5. Variable Selection. Use the t and p-values from summary(model)
6. Fitting. Re-run the model. Interpret the coefficients.
7. Prediction. Use the *predict.lm()* command.

Linear Regression Model

The Multiple Linear Regression (MLR) Model :

$$Y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \varepsilon$$

where ε represents the random error, in general $\varepsilon \sim N(0, \sigma^2)$.

- If $p = 1$, the linear model is called the **simple linear regression** model.
- β_i: If x_i increases by one unit holding the other x's constant, then Y will react by β_i units.
- Remark: there are p predictors here while $(p - 1)$ in the textbook.

Matrix Formulation of the Regression Model

The MLR model with observations is

$$Y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi} + \varepsilon_i \ (i = 1, 2, \ldots, n)$$

or

$$Y_i = \boldsymbol{x}_i^{'} \boldsymbol{\beta} + \varepsilon_i (i = 1, 2, \ldots, n)$$

where $\boldsymbol{x}_i^{'} = (1, x_{1i}, x_{2i}, \ldots, x_{pi})$.

The MLR model with matrix

$$\mathbf{Y}_{n\times 1} = \mathbf{X}_{n\times(p+1)} \boldsymbol{\beta}_{(p+1)\times 1} + \boldsymbol{\varepsilon}_{n\times 1}, \text{ where } \boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$$

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}, \ \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{p1} \\ 1 & x_{12} & x_{22} & \cdots & x_{p2} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{pn} \end{bmatrix} \ \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}$$

Model Assumptions

The models are valid underlying the following assumptions:

1. For given values of x, the Y values are normally distributed (Normality).
2. The $E(Y|x)$ is a linear function of $\boldsymbol{x} = (1, x_1, x_2, \dots, x_p)^{'}$ (Linearity) .
3. The standard deviations of Y_i are equal (Homoscedasticity).
4. The $Y_i (i = 1, 2, \dots, n)$ are statistically independent (Independence).
5. There's also the issue of outliers and influential observations.

1.2 Estimation

Least Squares and Maximum Likelihood Estimates

The **Least Squares Estimates** (LSE) of the unknown parameters minimize

$$\begin{aligned} Q(\boldsymbol{\beta}) &= \sum [y_i - (\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi})]^2 \\ &= (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^{'}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) \end{aligned}$$

The **Maximum Likelihhod Estimates** (MLE) maximize

$$L(\boldsymbol{\beta}, \sigma^2) = \frac{1}{(2\pi\sigma)^{n/2}} \exp\left\{\frac{1}{\sigma^2}[(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^{'}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})]\right\}$$

By differentiating $Q(\boldsymbol{\beta})$ and $L(\boldsymbol{\beta}, \sigma^2)$ with respect to $\boldsymbol{\beta}$ yields

$$(\mathbf{X}^{'}\mathbf{X})\boldsymbol{\beta} = \mathbf{X}^{'}\mathbf{Y}$$

Hence the estimator of $\boldsymbol{\beta}$ (LSE and MLE both) are

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^{'}\mathbf{X})^{-1}\mathbf{X}^{'}\mathbf{Y}$$

Properties of the coefficient estimators

1. The fitted or predicted values are given by

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}^{'}\mathbf{X})^{-1}\mathbf{X}^{'}\mathbf{Y} = \mathbf{H}\mathbf{Y}$$

   where $\mathbf{H} = \mathbf{X}(\mathbf{X}^{'}\mathbf{X})^{-1}\mathbf{X}^{'}$ is appropriately named the **hat matrix** because it "puts the hat on $\mathbf{Y}$".

2. $\hat{\boldsymbol{\beta}}$ is an unbiased estimate of $\boldsymbol{\beta}$, and

$$\hat{\boldsymbol{\beta}}|\mathbf{X} \sim N[\boldsymbol{\beta}, \sigma^2(\mathbf{X}^{'}\mathbf{X})^{-1}]$$

3. It can be shown that S^2 is an unbiased estimate of σ^2, which

$$S^2 = \frac{SSE}{n-p-1}$$

   where $SSE = \sum_i (y_i - \hat{y}_i)^2$ and $S = \sqrt{S^2}$ is called the **standard error**.

2 Model Assessment and Inference

2.1 Assessment

ANOVA Table

Source of Variation	Degree of Freedom	Sum of Squares	Mean Squares	F
Regression	p	SSR	$MSR = SSR/p$	$F = \frac{MSR}{MSE}$
Error	$n-p-1$	SSE	$MSE = \frac{SSE}{n-p-1}$	
Total	$n-1$	SST		

$$SST = \sum_i (y_i - \bar{y})^2$$

$$SSR = \sum_i (\hat{y}_i - \bar{y})^2$$

$$SSE = \sum_i (y_i - \hat{y}_i)^2$$

Goodness-of-Fit

- **Coefficient of determination**

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

- **Interpretation**: the proportion of the variance in the outcome variable that can be accounted for by the predictor
- **The adjusted R^2**

$$R^2_{adj} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

  where n is sample size and p is the number of predictors.
- **Advantage**: when add more predictors to the model, the adjusted R^2_{adj} value will only increase if the new variables improve the model performance more than you'd expect by chance.

F-test for the Significance of the Model

Testing the model as a whole (*Analysis of variance* when $p > 2$)
$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0$
H_1 : at least one of $\beta_1, \beta_2, \cdots, \beta_p$ is not 0
F-test:

$$F = \frac{SSR/p}{SSE/(n-p-1)} \overset{H_0}{\sim} F(p, n-p-1)$$

2.2 Inferences

t-test for the significance of coefficients

- Tests the significance for individual coefficients

$$H_0 : \beta_i = 0 \text{ against } H_{1:}\beta_i \neq 0 (i = 1, \ldots, p)$$

- t-test

$$T_i = \frac{\hat{\beta}_i - \beta_i}{\mathrm{SE}(\hat{\beta}_i)} \overset{H_0}{\sim} t(n-p-1)$$

where $\mathrm{SE}(\hat{\beta}_i)$ is the ith diagonal element of $s^2(\mathbf{X}'\mathbf{X})^{-1}$ (Note that $\hat{\boldsymbol{\beta}}|\mathbf{X} \sim N[\boldsymbol{\beta}, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}]$).

- *summary()*

Confidence Interval for the Regression Coefficient

A $(1-\alpha)100\%$ confidence interval for coefficient β_i is

$$\mathrm{CI}(\beta_i) = \hat{\beta}_i \pm t_{1-\alpha/2}(n-p-1)\mathrm{SE}(\hat{\beta}_i)$$

In R: *confint(model, level=0.95)*

Confidence and Prediction Intervals for a Future Value

The **confidence interval** for the mean value $\mathrm{E}(\mathbf{Y}|\mathbf{x}_0)$ of a future observation $\mathbf{x}_0 = (1, x_{10}, x_{20}, \cdots, x_{p0})'$ are given by

$$\mathbf{x}_0'\hat{\boldsymbol{\beta}} \pm t_{1-\alpha/2}(n-p-1)S\sqrt{\mathbf{x}_0'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_0}$$

The **prediction interval** for a new observation $\mathbf{Y}_0$ at $\mathbf{x}_0$ are given by

$$\mathbf{x}_0'\hat{\boldsymbol{\beta}} \pm t_{1-\alpha/2}(n-p-1)S\sqrt{1 + \mathbf{x}_0'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_0}$$

Example: Google Stock Returns

Consider a CAPM regression for Google's stock

$$\mathrm{Google}_t = \alpha + \beta \mathrm{sp500}_t + \epsilon_t$$

This is a market model.

- Alpha: The alpha of the portfolio measures the performance of a stock (a fund).
- Beta: The beta of a portfolio measures how sensitive the portfolio's return is to the movement of the overall market.

Question:

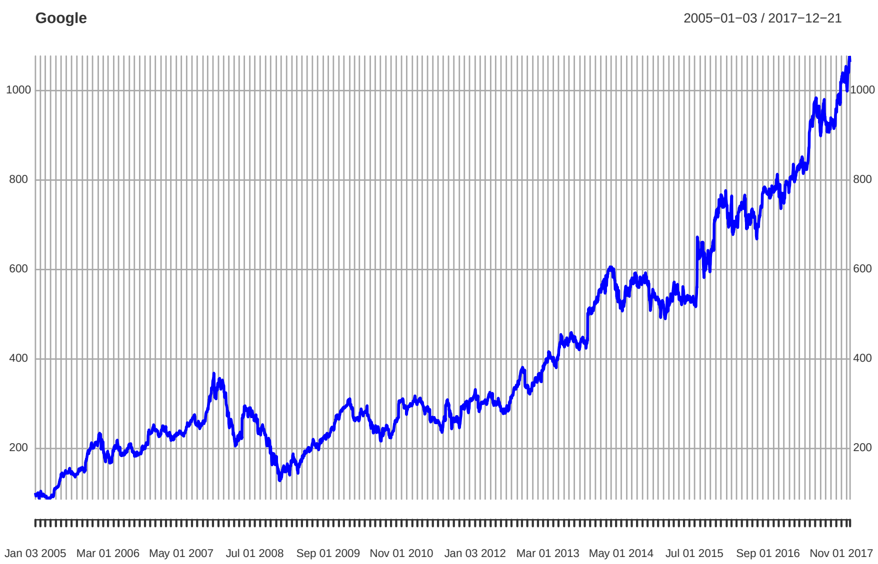
- is Google related to the market?
- Does Google out-perform the market in a consistent fashion?
- is Google better than Apple?

Example: Google data and log-returns

```
> library(quantmod)
> getSymbols('GOOG', from = "2005-01-01")

## [1] "GOOG"

> # Data from SPY, the ETF tracks SP500
> getSymbols('SPY', from = "2005-01-01")

## [1] "SPY"

> x <- SPY$SPY.Close
> y <- GOOG$GOOG.Close
> n <- length(y)
> ret <- diff(log(y))
> ret <- ret[-1]
> SP500ret <- diff(log(x))
> SP500ret <- SP500ret[-1]
```

Example: Google 2010-2017

```
> plot(y,type="l",col=20,main="Google",xlab="Price",ylab="$")
```



Example: CAPM Market Model

Example: CAPM Market Model

```
> googlemkt<-lm(ret~SP500ret); summary(googlemkt)

##
## Call:
## lm(formula = ret ~ SP500ret)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.109065 -0.006459 -0.000355  0.006182  0.172141
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0005020  0.0002635   1.905   0.0569 .
## SP500ret    0.9256294  0.0221102  41.864   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01499 on 3234 degrees of freedom
## Multiple R-squared:  0.3515,Adjusted R-squared:  0.3513
## F-statistic:  1753 on 1 and 3234 DF,  p-value: < 2.2e-16
```

Example: Prediction

Suppose the SP500's return will be 1% and 1.2% next week, You want to predict:

- The average return at that market return levels.
- The daily return of GOOGLE if the market returns are 1% and 1.2% respectively.

R Usage

After running `model=lm(`$y \sim x$`)`,

- Define a vector data for prediction

  `new=data.frame(x=c(1,1.2))`

- *Predicted interval: for one sample value*

  `predict.lm(model,new,interval="prediction")`

- *Predicted confidence interval: for average value*

  `predict.lm(model,new,interval="confidence")`

Predicting the Google Returns

```
> new <- data.frame(SP500ret = c(1, 1.2))
> predict.lm(googlemkt,new,interval="prediction")

##         fit       lwr       upr
## 1 0.9261314 0.8737656 0.9784972
## 2 1.1112573 1.0515169 1.1709976

> predict.lm(googlemkt,new,interval="confidence")

##         fit       lwr       upr
## 1 0.9261314 0.8827871 0.9694757
## 2 1.1112573 1.0592432 1.1632713
```

3 Fitting Regression Models in R

3.1 Murder rate in USA

States dataset

The *state.x77* in the dataset package is a *matrix* with 8 columns giving statistics for the states:

1. Population estimate as of July 1, 1975;
2. Per capita Income (1974);
3. Illiteracy (1970, percent of population);
4. Life Expectancy in years (1969-71);
5. Murder and non-negligent manslaughter rate per 100,000 population (1976);
6. Percent High-school Graduates (1970);
7. Mean Number of days with min temperature below freezing (1931-1960) in capital or large city;
8. Land Area in square miles.

Goal: explore the relationship between a state's **murder** rate and other characteristics , including **population**, **illiteracy** rate, average **income**, and **frost** levels

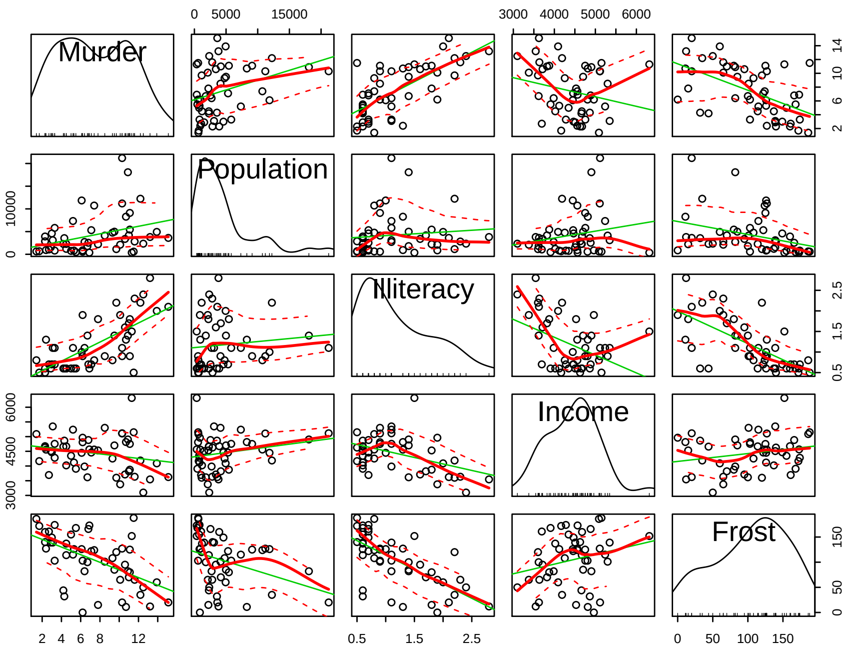
Correlations and scatter plots

Extract the relative variables from state.x77 as a *data frame*:

```
> library("datasets")
> states<-as.data.frame(state.x77[,c("Murder","Population",
+                       "Illiteracy","Income","Frost")])
> round(cor(states),3)

##            Murder Population Illiteracy Income  Frost
## Murder      1.000      0.344      0.703 -0.230 -0.539
## Population  0.344      1.000      0.108  0.208 -0.332
## Illiteracy  0.703      0.108      1.000 -0.437 -0.672
## Income     -0.230      0.208     -0.437  1.000  0.226
## Frost      -0.539     -0.332     -0.672  0.226  1.000
```

scatterplotMatrix( ) in the car package

```
> library(car); scatterplotMatrix(states)
```



Some descriptive conclusions

- Murder rate may be bimodal
- Each of the predictor variables is skewed to some extent
- Murder rates rise with population and illiteracy
- Murder rates fall with higher income levels and frost

- Colder states have lower illiteracy rates
- Colder states have lower population and higher incomes

Fit the model with all variables

```
> fit<-lm(Murder~.,data = states); summary(fit)

##
## Call:
## lm(formula = Murder ~ ., data = states)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7960 -1.6495 -0.0811  1.4815  7.6210
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.235e+00  3.866e+00   0.319   0.7510
## Population  2.237e-04  9.052e-05   2.471   0.0173 *
## Illiteracy  4.143e+00  8.744e-01   4.738 2.19e-05 ***
## Income      6.442e-05  6.837e-04   0.094   0.9253
## Frost       5.813e-04  1.005e-02   0.058   0.9541
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.535 on 45 degrees of freedom
## Multiple R-squared:  0.567,Adjusted R-squared:  0.5285
## F-statistic: 14.73 on 4 and 45 DF,  p-value: 9.133e-08
```

anova(model)

```
> anova(fit)

## Analysis of Variance Table
##
## Response: Murder
##            Df  Sum Sq Mean Sq F value   Pr(>F)
## Population  1  78.854  78.854 12.2713 0.001052 **
## Illiteracy  1 299.646 299.646 46.6307 1.83e-08 ***
## Income      1   0.057   0.057  0.0089 0.925368
## Frost       1   0.021   0.021  0.0033 0.954148
## Residuals  45 289.167   6.426
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Fit the model with all variables

```
> library("xtable")
> xtable(summary(fit))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.2346	3.8661	0.32	0.7510
Population	0.0002	0.0001	2.47	0.0173
Illiteracy	4.1428	0.8744	4.74	0.0000
Income	0.0001	0.0007	0.09	0.9253
Frost	0.0006	0.0101	0.06	0.9541

Interpretations

- The coefficients for **Illiteracy** and **Population** are significantly different from zero
- The coefficient for **Illiteracy** is 4.14, suggesting that an increase of 1 percent in illiteracy is associated with a 4.14 percent increase in the murder rate, controlling for population, income, and temperature
- The coefficient for **Population** is 0.0002, suggesting that the population is statistical significant but isn't economic significant
- **Frost** and **Income** are't significant, suggesting that **Frost, Income** and **Murder** are't linearly related when controlling for the other predictor variables
- $R^2 = 57\%$: taken all the predictor variables together, account for 57 percent of the variance in murder rates across states

Fit the model with significant variables Only

```
> fit2<-lm(Murder~Population+Illiteracy, data = states)
> summary(fit2)

##
## Call:
## lm(formula = Murder ~ Population + Illiteracy, data = states)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7652 -1.6561 -0.0898  1.4570  7.6758
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.652e+00  8.101e-01   2.039  0.04713 *
## Population  2.242e-04  7.984e-05   2.808  0.00724 **
## Illiteracy  4.081e+00  5.848e-01   6.978 8.83e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.481 on 47 degrees of freedom
## Multiple R-squared:  0.5668,Adjusted R-squared:  0.5484
## F-statistic: 30.75 on 2 and 47 DF,  p-value: 2.893e-09
```

3.2 Selecting the "best" regression model

3.2.1 Testing-Based Procedures

Stepwise regression

- In **stepwise** selection, variables are added to or deleted from a model one at a time, until some stopping criterion is reached
- **Forward stepwise regression** you add predictor variables to the model one at a time
- **Backward stepwise regression** you start with a model that includes all predictor variables, and then delete them one at a time until removing variables would degrade the quality of the model
- *step( )* function or *stepAIC( )* function in the MASS package

Backward stepwise selection

```
> fit<-lm(Murder~., data=states)
> step(fit,direction="backward")

## Start:  AIC=97.75
## Murder ~ Population + Illiteracy + Income + Frost
##
##              Df Sum of Sq    RSS     AIC
## - Frost       1     0.021 289.19  95.753
## - Income      1     0.057 289.22  95.759
## <none>                    289.17  97.749
## - Population  1    39.238 328.41 102.111
## - Illiteracy  1   144.264 433.43 115.986
##
## Step:  AIC=95.75
## Murder ~ Population + Illiteracy + Income
##
##              Df Sum of Sq    RSS     AIC
## - Income      1     0.057 289.25  93.763
## <none>                    289.19  95.753
## - Population  1    43.658 332.85 100.783
## - Illiteracy  1   236.196 525.38 123.605
##
## Step:  AIC=93.76
## Murder ~ Population + Illiteracy
##
##              Df Sum of Sq    RSS     AIC
## <none>                    289.25  93.763
## - Population  1    48.517 337.76  99.516
## - Illiteracy  1   299.646 588.89 127.311
##
## Call:
## lm(formula = Murder ~ Population + Illiteracy, data = states)
##
## Coefficients:
## (Intercept)   Population   Illiteracy
##   1.6515497    0.0002242    4.0807366
```

3.2.2 Criterion-Based Procedures

Criteria

- R^2_{adj}:

$$R^2_{adj} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

- Mallows's C_p: if p regressors are selected from a set of $k > p$, C_p for that particular set of regressors is defined as:
$$C_p = \frac{SSE_p}{S^2} + 2p - n$$
Desirable models have small p and C_p less than or equal to p.

Criteria

- Bayesian Information Criterion (BIC):
$$\begin{aligned} BIC &= -2\ln L(\hat{\theta}_{MLE}|\boldsymbol{x}) + (p+1)\ln(n) \\ &= n\ln(SSE/n) + (p+1)\ln(n) + \text{constant} \end{aligned}$$
Models with smaller BIC values indicating preferred.
- Akaike Information Criterion (AIC):
$$AIC = -2\ln L(\hat{\theta}_{MLE}|\boldsymbol{x}) + 2(p+1)$$
BIC will favor smaller models than will AIC (BIC penalizes larger models as AIC (assuming $n > e^2 = 7.3891$).

Model comparison using anova()

anova(): can be used to compare nested models

Nested model: one whose terms are completely included in the other model

```
> fit1 <- lm(Murder ~ . , data=states)
> fit2 <- lm(Murder~Population+Illiteracy,data=states)
> anova(fit2,fit1)

## Analysis of Variance Table
##
## Model 1: Murder ~ Population + Illiteracy
## Model 2: Murder ~ Population + Illiteracy + Income + Frost
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     47 289.25
## 2     45 289.17  2  0.078505 0.0061 0.9939
```

Conclusion: the Income and Frost are nonsignificant (p = .994), should be dropped from our model

Model Comparison Using Information Criterion

```
> AIC(fit2,fit1)

##      df      AIC
## fit2  4 237.6565
## fit1  6 241.6429

> BIC(fit2,fit1)

##      df      BIC
## fit2  4 245.3046
## fit1  6 253.1151
```

Conclusion: the AIC values suggest that the model without Income and Frost is the better model

All subsets regression

In **all subsets regression**, every possible model is inspected
regsubsets( ) function in the leaps package
Model selecting criteria:

- Adjusted R-squared
- Mallows Cp statistic
- BIC

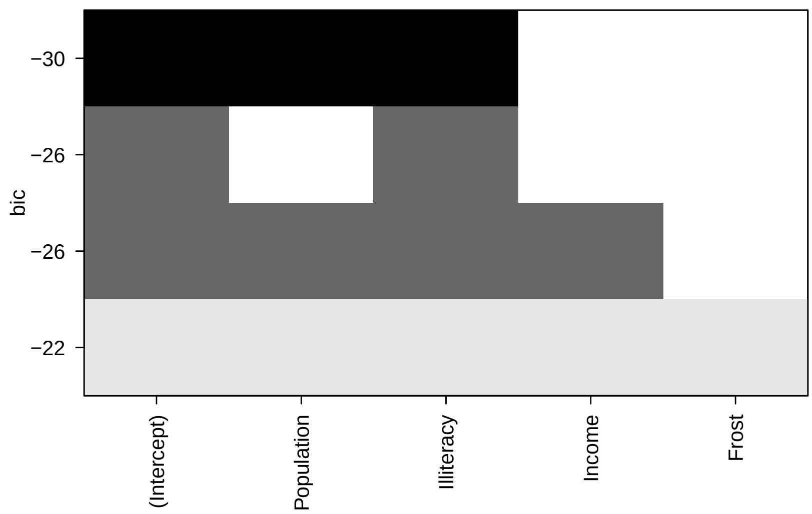
plot( ) function in the leaps package

Run the best subset regression

```
> library(leaps)
> fitsub<-regsubsets(Murder~ . , data=states)
> summary(fitsub)

## Subset selection object
## Call: regsubsets.formula(Murder ~ ., data = states)
## 4 Variables  (and intercept)
##            Forced in Forced out
## Population     FALSE      FALSE
## Illiteracy     FALSE      FALSE
## Income         FALSE      FALSE
## Frost          FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: exhaustive
##          Population Illiteracy Income Frost
## 1  ( 1 ) " "        "*"        " "    " "
## 2  ( 1 ) "*"        "*"        " "    " "
## 3  ( 1 ) "*"        "*"        "*"    " "
## 4  ( 1 ) "*"        "*"        "*"    "*"
```

Display the best subset by plot() in leaps

```
> plot(fitsub,scale="bic")
```



What's your conclusions?