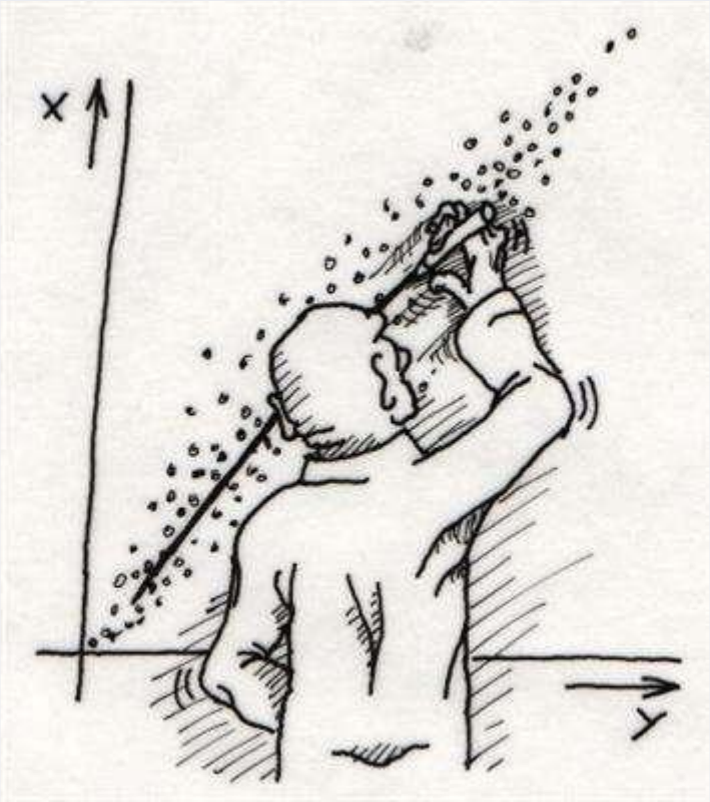


Lecture 5: Multiple Linear Regression

Pilsung Kang
School of Industrial Management Engineering
Korea University

AGENDA

Multiple Linear Regression

- Regression Example: Predict the selling price of Toyota Corolla


Dependent variable (target): Price

Independent variables (attributes, features): Age_08_04, KM, Fuel_Type, HP, Met_Color, Automatic, CC, Doors, Quarterly_Tax, Weight

Variable	Description
Price	Offer Price in EUROs
Age_08_04	Age in months as in August 2004
KM	Accumulated Kilometers on odometer
Fuel_Type	Fuel Type (Petrol, Diesel, CNG)
HP	Horse Power
Met_Color	Metallic Color? (Yes=1, No=0)
Automatic	Automatic ((Yes=1, No=0)
CC	Cylinder Volume in cubic centimeters
Doors	Number of doors
Quarterly_Tax	Quarterly road tax in EUROs
Weight	Weight in Kilograms

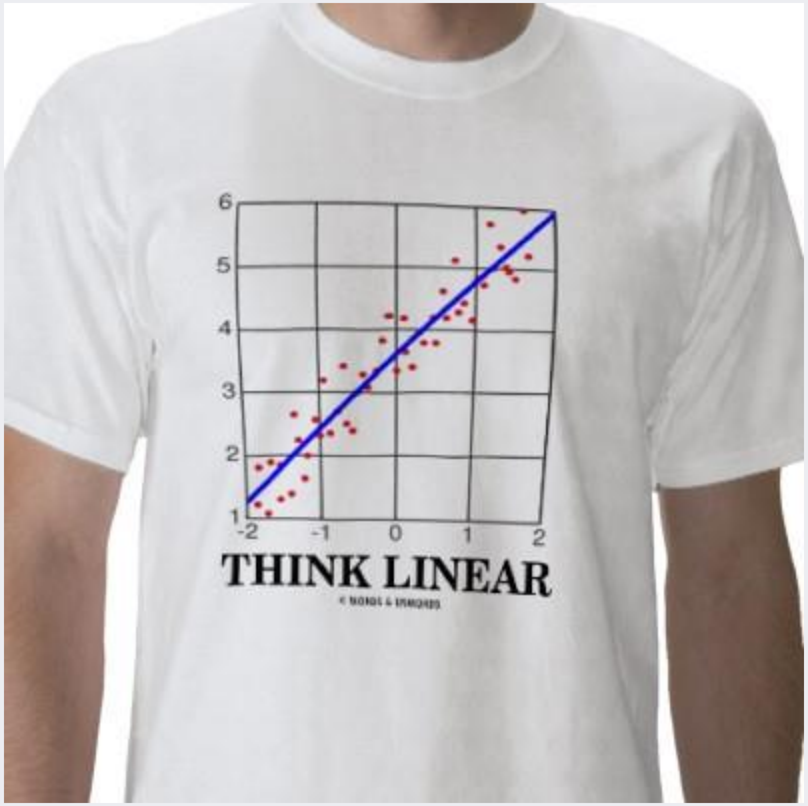
Multiple Linear Regression

- Goal
  - ✓ Fit a linear relationship between a quantitative dependent variable Y and a set of predictors $X_1, X_2, \ldots, X_p$.

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \varepsilon$$

coefficients

unexplained



Multiple Linear Regression

- Explanatory vs. Predictive

Explanatory Regression

- Explain relationship between predictors (explanatory variables) and target.
- Familiar use of regression in data analysis.
- Model Goal: Fit the data well and understand the contribution of explanatory variables to the model.
- "goodness-of-fit": R^2, residual analysis, p-values.

$$Y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + \varepsilon$$

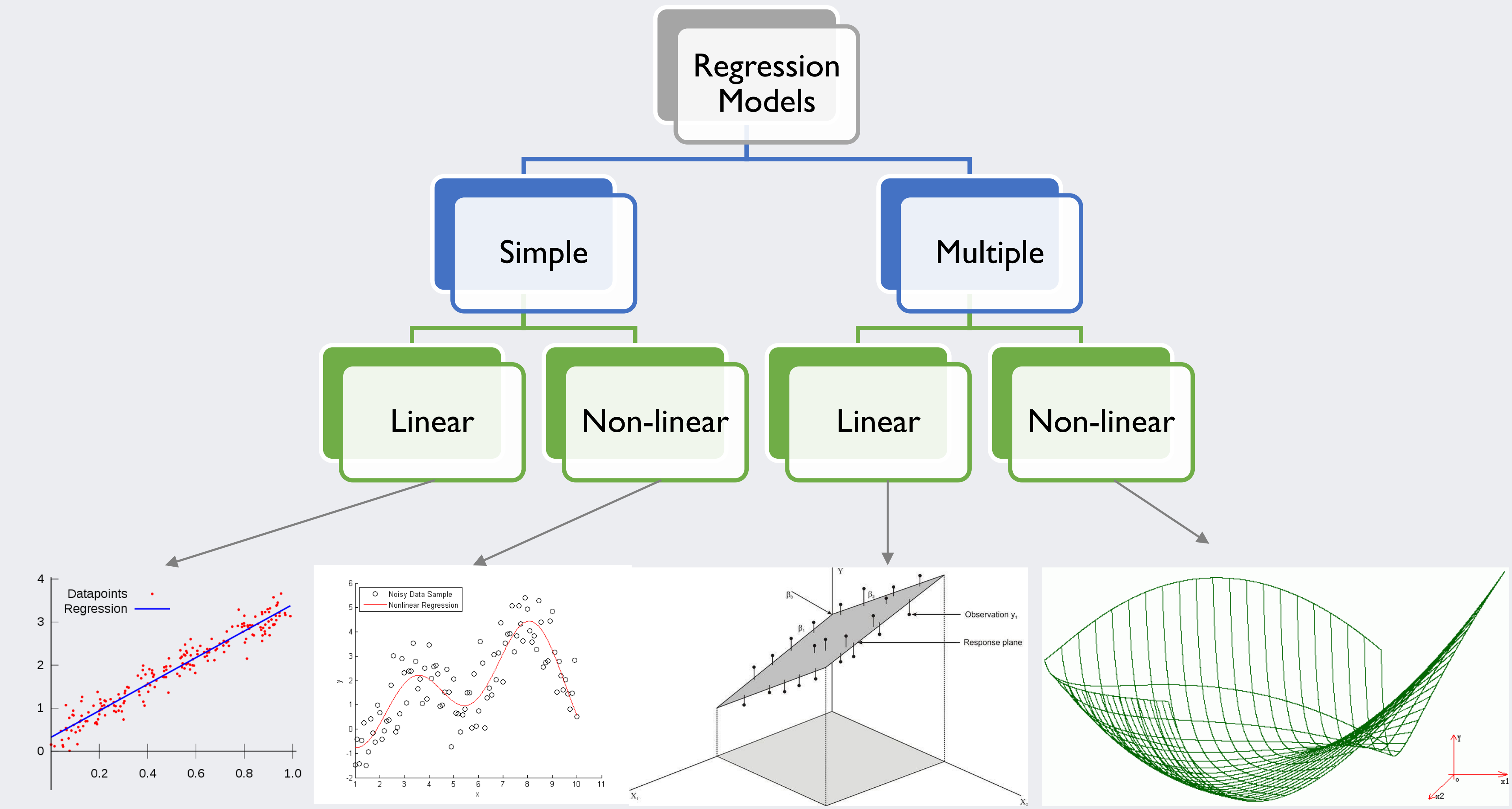
Predictive Regression

- Predict target values in other data where we have predictor values, but not target values.
- Classic data mining context
- Model Goal: Optimize predictive accuracy
- Train model on training data
- Assess performance on validation (hold-out) data
- Explaining role of predictors is not primary purpose (but useful)

$$Y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + \varepsilon$$

Multiple Linear Regression

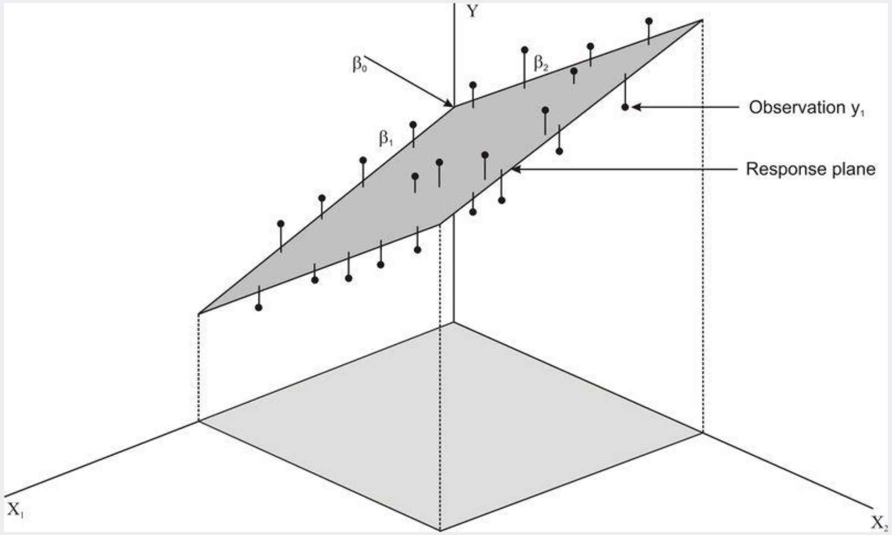
- Type of Regression



Multiple Linear Regression

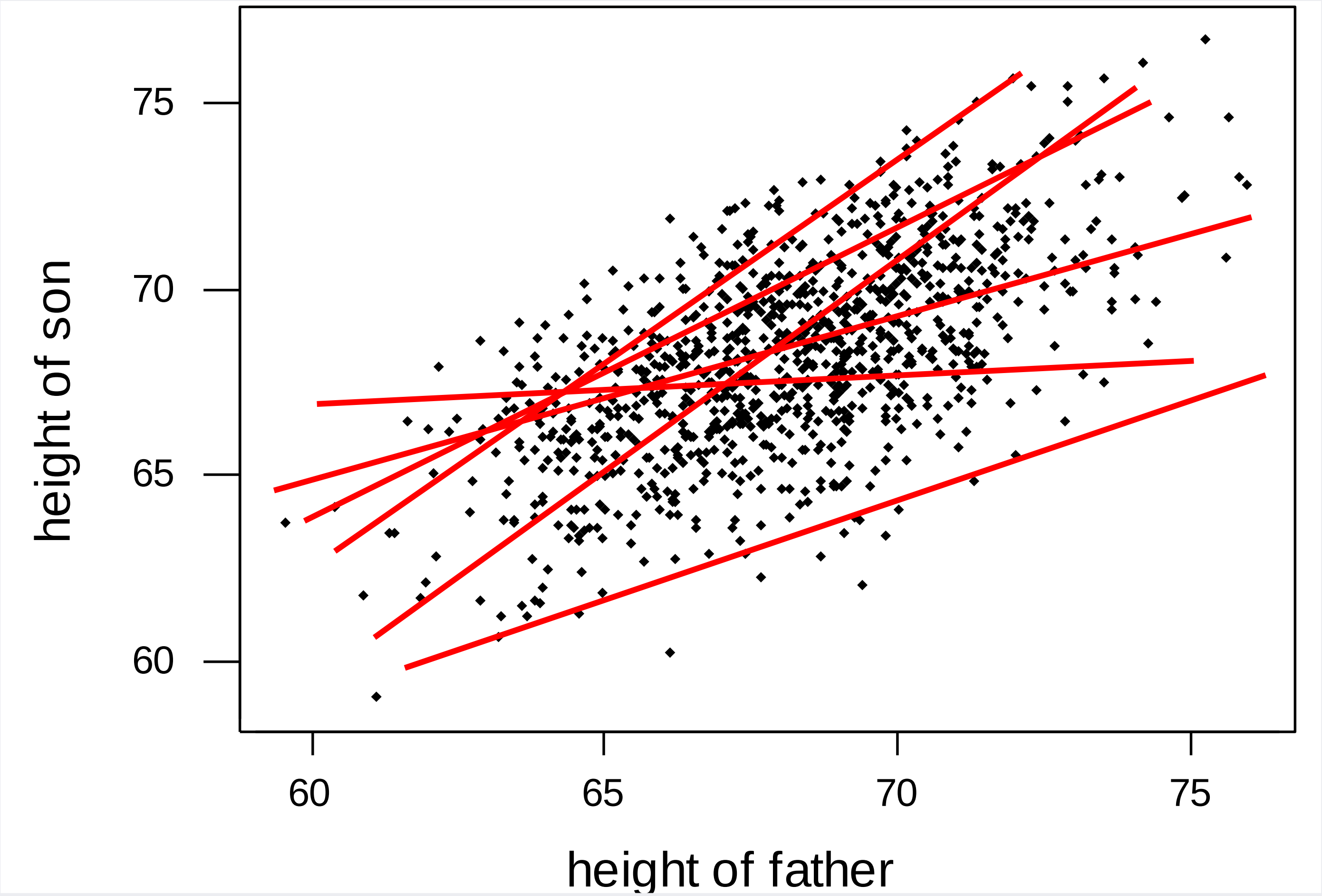
- Linear Regression
  - ✓ Assume that the relationship between the input variable and the target variable is always linear.

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \varepsilon$$



Multiple Linear Regression

- Which line is optimal?



Multiple Linear Regression

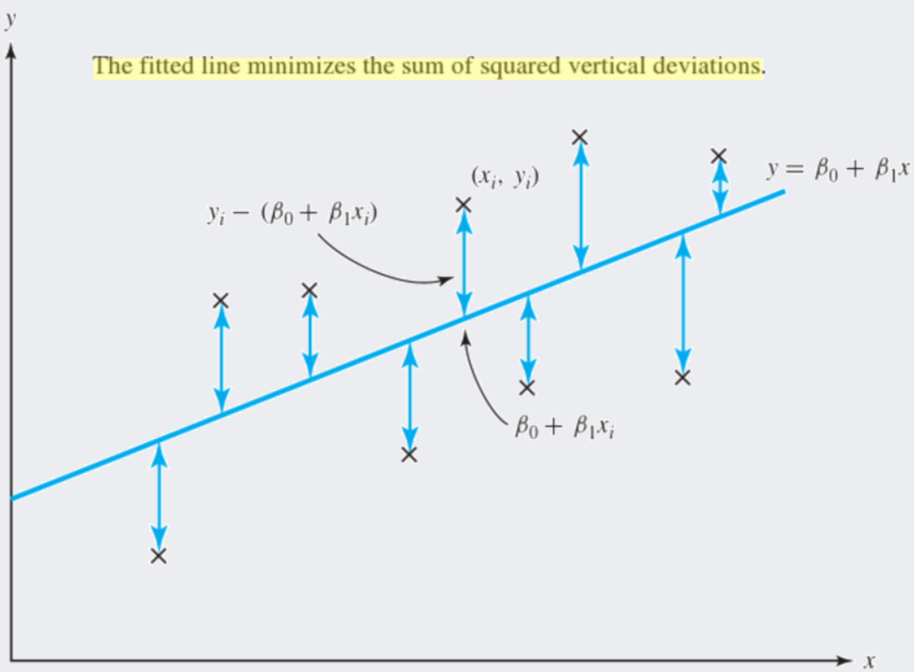
- Estimating the coefficients
  - ✓ Ordinary least square (OLS)
    - Actual target: $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \varepsilon$
    - Predicted target: $\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \ldots + \hat{\beta}_p x_p$
    - Goal: minimize the difference between the actual and predicted target.

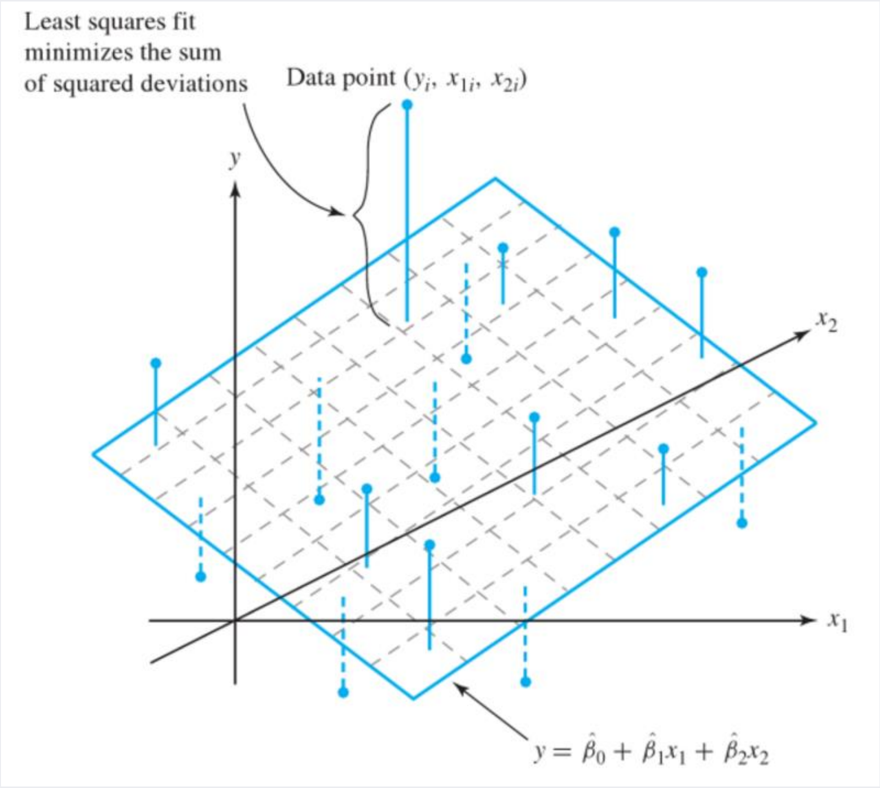
$$\min \quad \frac{1}{2}\sum_{i=1}^{N} \varepsilon_i^2 = \frac{1}{2}(Y_i - \hat{Y}_i)^2$$

$$= \frac{1}{2}\left(Y - \hat{\beta}_0 - \hat{\beta}_1 x_1 - \hat{\beta}_2 x_2 - \ldots - \hat{\beta}_p x_p\right)^2$$

Multiple Linear Regression

- Estimating the coefficients
    - ✓ Ordinary least square (OLS)





Multiple Linear Regression

- Ordinary least square: Matrix solution
  - ✓ **X**: n by p matrix, **y**: n by 1 vector, **β**: p by 1 vector.

$$\min \quad E(\mathbf{X}) = \frac{1}{2}\left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\right)^{\mathrm{T}}\left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\right)$$

$$\Rightarrow \quad \frac{\partial E(\mathbf{X})}{\partial \boldsymbol{\beta}} = -\left(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\right)^{\mathrm{T}} \mathbf{X} = 0$$

$$\Rightarrow \quad -\mathbf{y}^{\mathrm{T}}\mathbf{X} + \boldsymbol{\beta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X} = 0$$

$$\Rightarrow \quad \boldsymbol{\beta}^{\mathrm{T}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{y}^{\mathrm{T}}\mathbf{X}$$

$$\Rightarrow \quad \boldsymbol{\beta} = \left((\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{y}^{\mathrm{T}}\mathbf{X}\right)^{\mathrm{T}}$$

Multiple Linear Regression

- Ordinary least square
  - ✓ Finds the best estimates **β** when the following conditions are satisfied:
    - The noise ε follows a normal distribution.
    - The linear relationship is correct.
    - The cases are independent of each other.
    - The variability in Y values for a given set of predictors is the same regardless of the values of the predictors (homoskedasticity).

Multiple Linear Regression

- Goodness-of-fit: (Adjusted) R^2

$$R^2 = 1 - \frac{\sum_{j=1}^{n} \hat{\varepsilon}_j^2}{\sum_{j=1}^{n} (y_j - \bar{y})^2} = \frac{\sum_{j=1}^{n} (\hat{y}_j - \bar{y})^2}{\sum_{j=1}^{n} (y_j - \bar{y})^2}$$

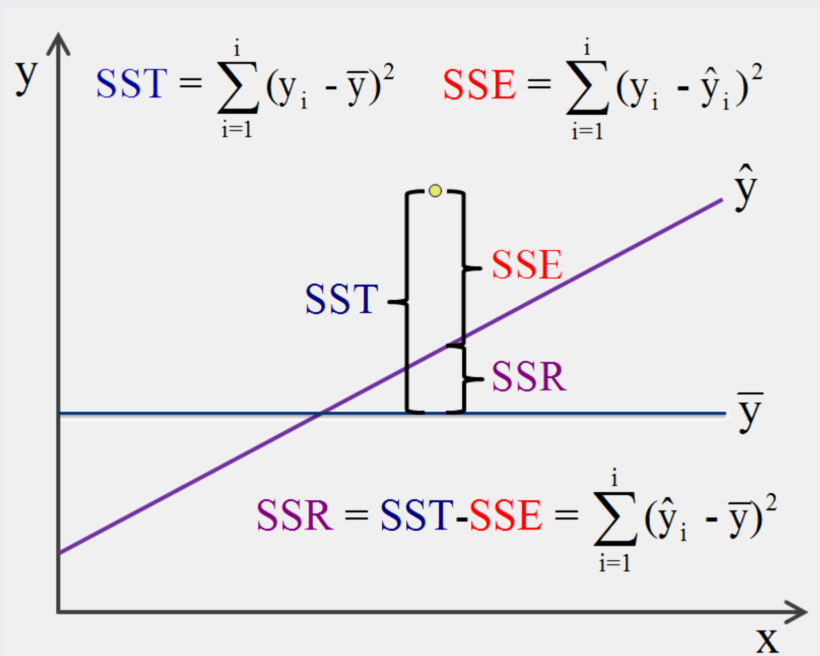
  - ✓ Gives the proportion of the total variation in the y_i's explained by the predictor variables
  - ✓ R^2 equals 1 if the fitted equation passes through all the data points

Multiple Linear Regression

- Sum-of-Squares Decomposition

$$\underbrace{\sum_{j=1}^{n}\left(y_j-\bar{y}\right)^2}_{\left(\substack{\text{total sum of squares}\\ \text{about mean}}\right)} = \underbrace{\sum_{j=1}^{n}\left(\hat{y}_j-\bar{y}\right)^2}_{\left(\substack{\text{regression}\\ \text{sum of squares}}\right)} + \underbrace{\sum_{j=1}^{n}\hat{\varepsilon}_j^2}_{\left(\substack{\text{residual (error)}\\ \text{sum of squares}}\right)}.$$

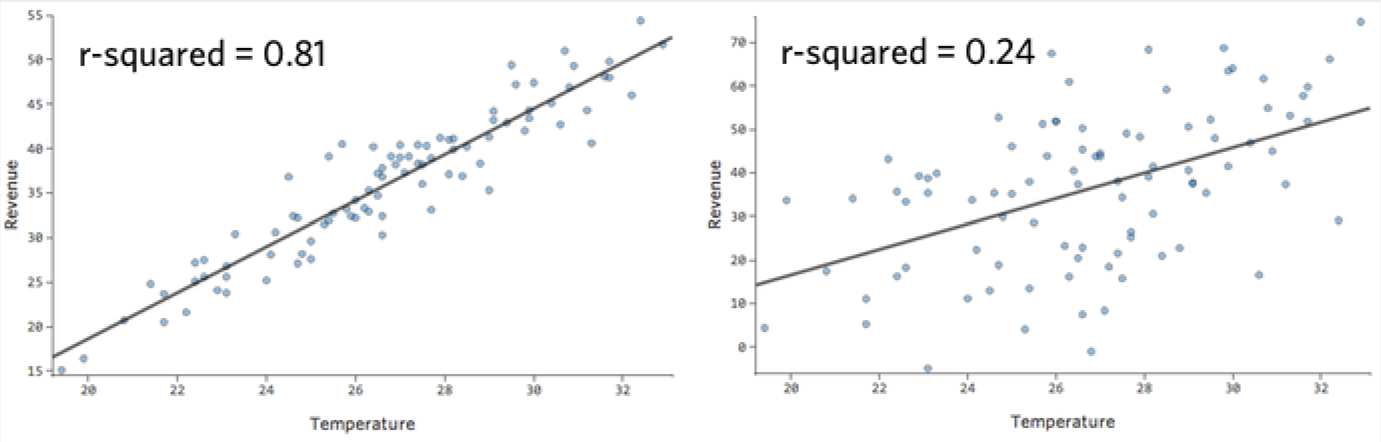
SST **SSR** **SSE**



Multiple Linear Regression

- Coefficient of Determination
  - ✓ The proportionate reduction of total variation associated with the use of the predictor variable Z.

$$R^2 = 1 - \frac{SSE}{SST} = \frac{SSR}{SST} \qquad 0 \leq R^2 \leq 1$$



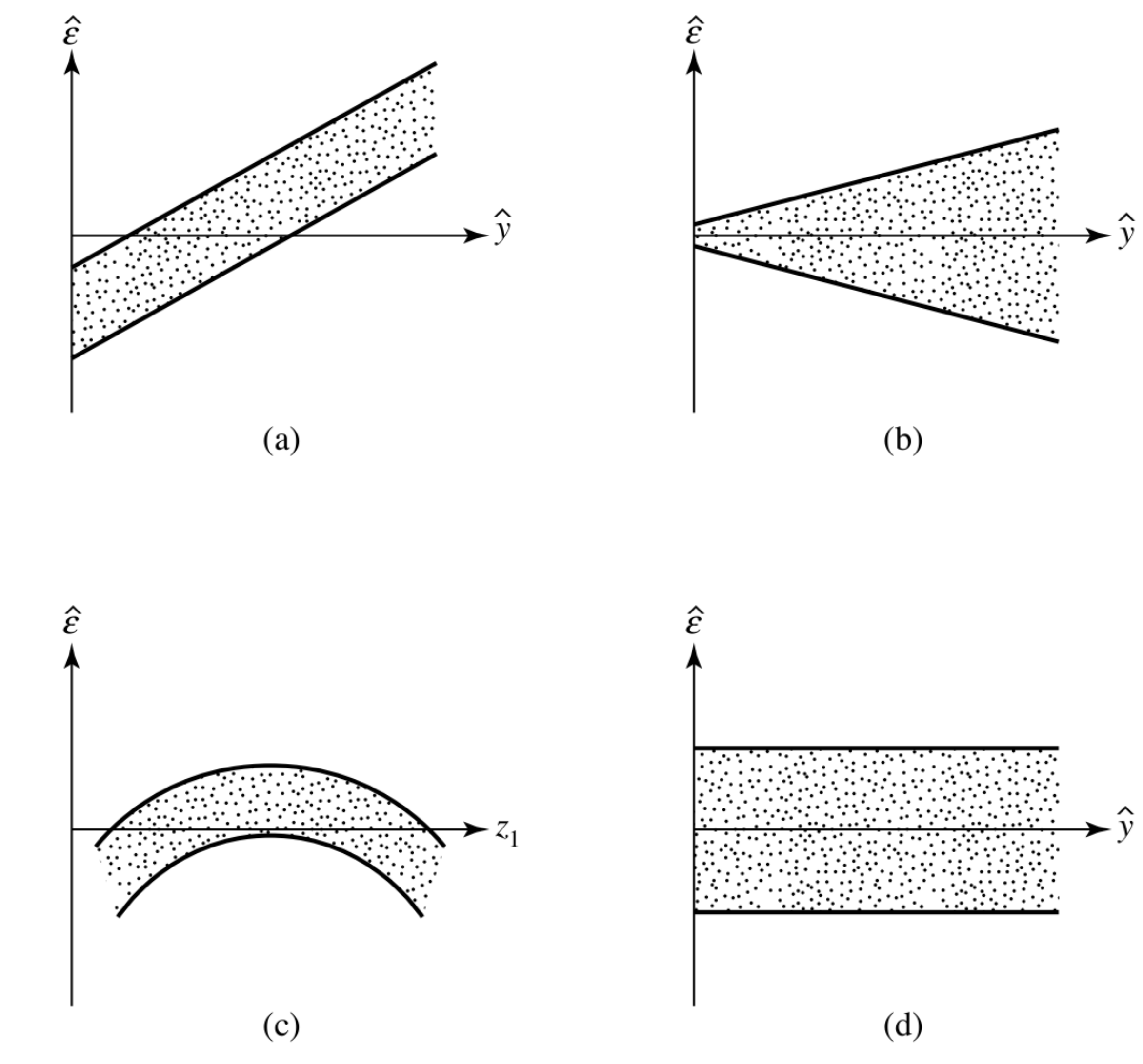
Multiple Linear Regression

- Model Fit
  - ✓ It is imperative to examine the adequacy of the model before the estimated function becomes a permanent part of the decision making apparatus.
  - ✓ For general diagnostic purpose, residuals should be plotted as follows:

1. *Plot the residuals $\hat{\varepsilon}_j$ against the predicted values $\hat{y}_j = \hat{\beta}_0 + \hat{\beta}_1 z_{j1} + \cdots + \hat{\beta}_r z_{jr}$.* Departures from the assumptions of the model are typically indicated by two types of phenomena:
2. *Plot the residuals $\hat{\varepsilon}_j$ against a predictor variable, such as z_1, or products of predictor variables, such as z_1^2 or $z_1 z_2$.* A systematic pattern in these plots suggests the need for more terms in the model. This situation is illustrated in Figure 7.2(c).
3. *Q–Q plots and histograms.* Do the errors appear to be normally distributed? To answer this question, the residuals $\hat{\varepsilon}_j$ or $\hat{\varepsilon}_j^*$ can be examined using the techniques discussed in Section 4.6. The *Q–Q* plots, histograms, and dot diagrams help to detect the presence of unusual observations or severe departures from normality that may require special attention in the analysis. If n is large, minor departures from normality will not greatly affect inferences about $\boldsymbol{\beta}$.

Multiple Linear Regression

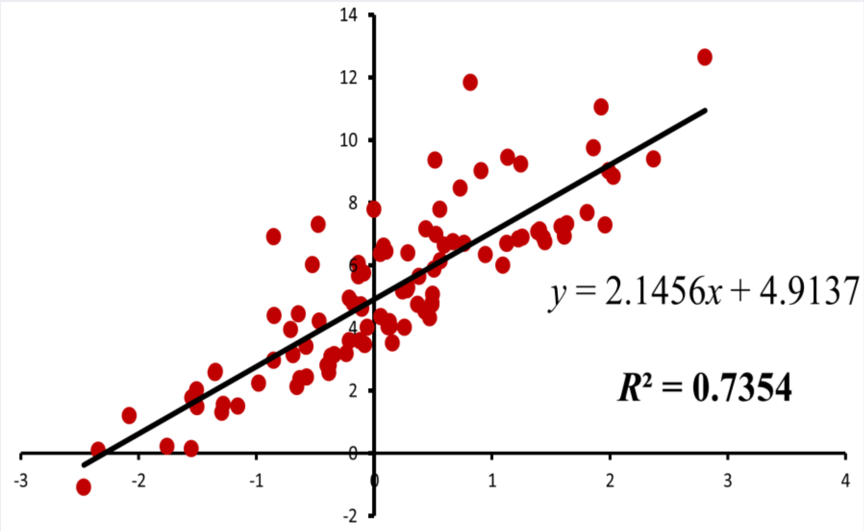
- Residual plots



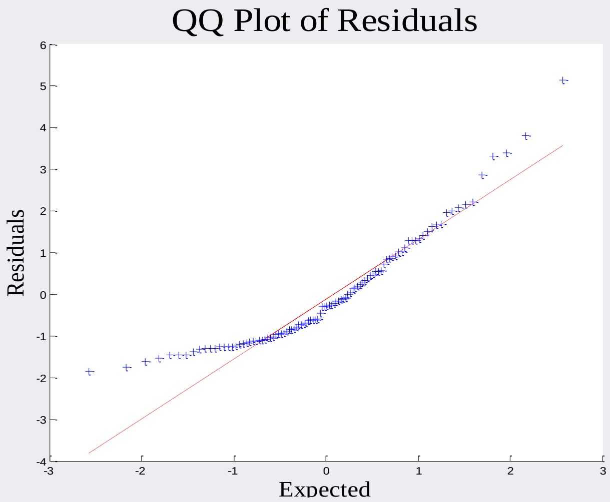
Multiple Linear Regression

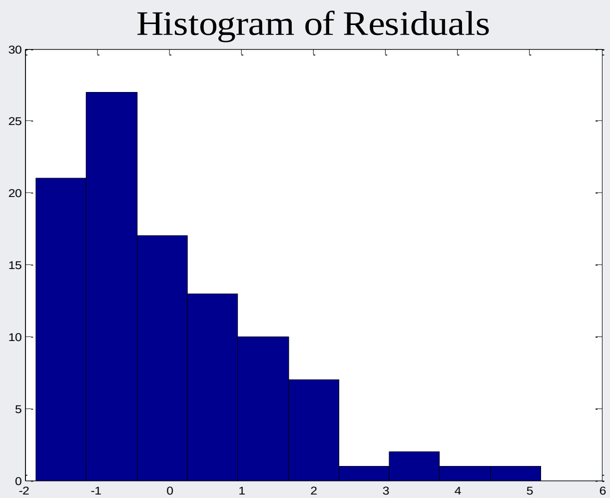
- Model checking

$$y = 2x + \varepsilon, \qquad \varepsilon \sim Gamma(2,1)$$



Regression model





Multiple Linear Regression: Example

- Example: predict the selling price of Toyota corolla

Y

X

Price	Age_08_04	KM	Fuel_Type	HP	Met_Color	Automatic	cc	Doors	Quarterly_Tax	Weight
13500	23	46986	Diesel	90	1	0	2000	3	210	1165
13750	23	72937	Diesel	90	1	0	2000	3	210	1165
13950	24	41711	Diesel	90	1	0	2000	3	210	1165
14950	26	48000	Diesel	90	0	0	2000	3	210	1165
13750	30	38500	Diesel	90	0	0	2000	3	210	1170
12950	32	61000	Diesel	90	0	0	2000	3	210	1170
16900	27	94612	Diesel	90	1	0	2000	3	210	1245
18600	30	75889	Diesel	90	1	0	2000	3	210	1245
21500	27	19700	Petrol	192	0	0	1800	3	100	1185
12950	23	71138	Diesel	69	0	0	1900	3	185	1105
20950	25	31461	Petrol	192	0	0	1800	3	100	1185
19950	22	43610	Petrol	192	0	0	1800	3	100	1185
19600	25	32189	Petrol	192	0	0	1800	3	100	1185
21500	31	23000	Petrol	192	1	0	1800	3	100	1185
22500	32	34131	Petrol	192	1	0	1800	3	100	1185
22000	28	18739	Petrol	192	0	0	1800	3	100	1185
22750	30	34000	Petrol	192	1	0	1800	3	100	1185
17950	24	21716	Petrol	110	1	0	1600	3	85	1105
16750	24	25563	Petrol	110	0	0	1600	3	19	1065

Multiple Linear Regression: Example

- Data preprocessing
  - ✓ Create dummy variables for fuel types

	Fuel_type = Disel	Fuel_type = Petrol	Fuel_type = CNG
Diesel	1	0	0
Petrol	0	1	0
CNG	0	0	1

- Data partitioning
  - ✓ 60% training data / 40% validation data

Id	Model	Price	Age_08_04	Mfg_Month	Mfg_Year	KM	Fuel_Type_Diesel	Fuel_Type_Petrol
1	RRA 2/3-Doors	13500	23	10	2002	46986	1	0
4	RRA 2/3-Doors	14950	26	7	2002	48000	1	0
5	SOL 2/3-Doors	13750	30	3	2002	38500	1	0
6	SOL 2/3-Doors	12950	32	1	2002	61000	1	0
9	/VT I 2/3-Doors	21500	27	6	2002	19700	0	1
10	RRA 2/3-Doors	12950	23	10	2002	71138	1	0
12	BNS 2/3-Doors	19950	22	11	2002	43610	0	1
17	ORT 2/3-Doors	22750	30	3	2002	34000	0	1

Multiple Linear Regression: Example

- Fitted linear regression model

Input variables	Coefficient	Std. Error	p-value	SS
Constant term	-3608.418457	1458.620728	0.0137	97276410000
Age_08_04	-123.8319168	3.367589	0	8033339000
KM	-0.017482	0.00175105	0	251574500
Fuel_Type_Diesel	210.9862518	474.9978333	0.6571036	6212673
Fuel_Type_Petrol	2522.066895	463.6594238	0.00000008	4594.9375
HP	20.71352959	4.67398977	0.00001152	330138600
Met_Color	-50.48505402	97.85591125	0.60614568	596053.75
Automatic	178.1519013	212.0528565	0.40124047	19223190
cc	0.01385481	0.09319961	0.88188446	1272449
Doors	20.02487946	51.0899086	0.69526076	39265060
Quarterly_Tax	16.7742424	2.09381151	0	160667200
Weight	15.41666317	1.40446579	0	214696000

β

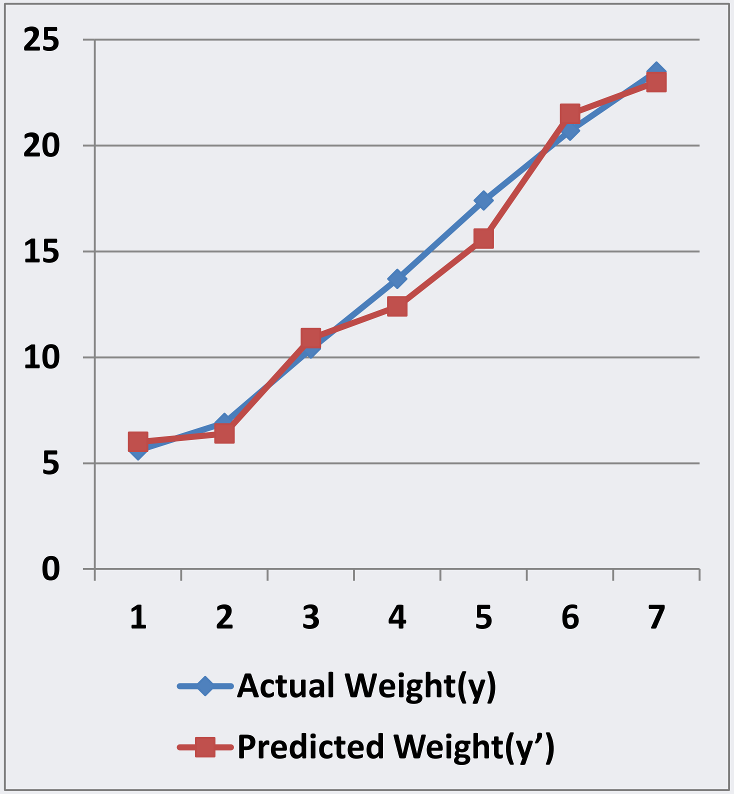
Significance Probability

AGENDA

Evaluating Regression Models

- Example: predict a baby's weight (kg) based on his/her age

Age	Actual Weight(y)	Predicted Weight(y')
1	5.6	6.0
2	6.9	6.4
3	10.4	10.9
4	13.7	12.4
5	17.4	15.6
6	20.7	21.5
7	23.5	23.0



Evaluating Regression Models

- Average error
  - ✓ Indicate whether the predictions are on average over- or under-predicted.

Age	Actual Weight(y)	Predicted Weight(y’)
1	5.6	6.0
2	6.9	6.4
3	10.4	10.9
4	13.7	12.4
5	17.4	15.6
6	20.7	21.5
7	23.5	23.0

$$Average\,error = \frac{1}{n}\sum_{i=1}^{n}(y - y')$$
$$= 0.342$$

Evaluating Regression Models

- Mean absolute error (MAE)
  - ✓ Gives the magnitude of the average error

Age	Actual Weight(y)	Predicted Weight(y')
1	5.6	6.0
2	6.9	6.4
3	10.4	10.9
4	13.7	12.4
5	17.4	15.6
6	20.7	21.5
7	23.5	23.0

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y - y'|$$
$$= 0.829$$

Evaluating Regression Models

- Mean absolute percentage error (MAPE)
  - ✓ Gives a percentage score of how predictions deviate (on average) from the actual values.

Age	Actual Weight(y)	Predicted Weight(y')
1	5.6	6.0
2	6.9	6.4
3	10.4	10.9
4	13.7	12.4
5	17.4	15.6
6	20.7	21.5
7	23.5	23.0

$$MAPE = 100\% \times \frac{1}{n} \sum_{i=1}^{n} \frac{|y - y'|}{|y|}$$

$$= 6.43\%$$

Evaluating Regression Models

- (Root) Mean squared error ((R)MSE)
  - ✓ Standard error of estimate
  - ✓ Same units as the variable predicted

Age	Actual Weight(y)	Predicted Weight(y’)
1	5.6	6.0
2	6.9	6.4
3	10.4	10.9
4	13.7	12.4
5	17.4	15.6
6	20.7	21.5
7	23.5	23.0

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y - y')^2$$
$$= 0.926$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y - y')^2}$$
$$= 0.962$$

AGENDA

01 Multiple Linear Regression

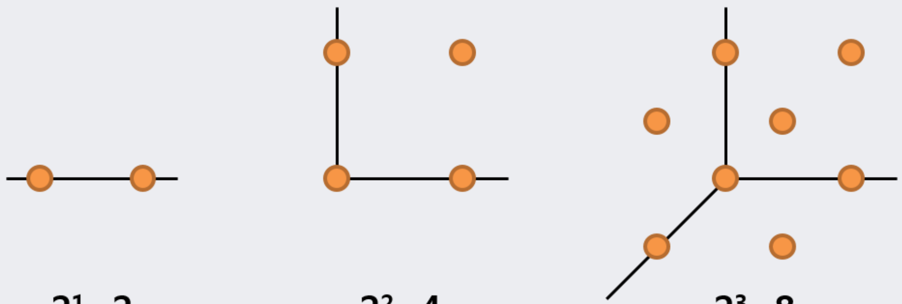
02 Evaluating Regression Models

03 Variable Selection

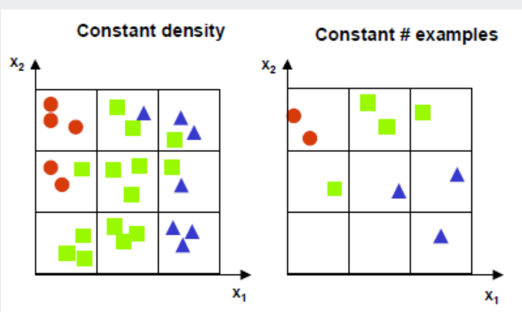
04 R Exercise

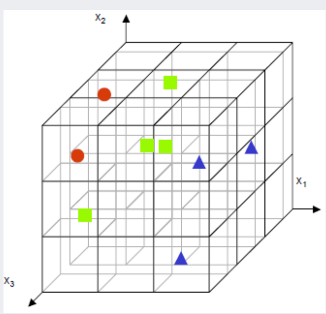
Variable Selection

- Curse of Dimensionality
  - ✓ The number of instances increases exponentially to achieve the same explanation ability when the number of variables increases



$2^1=2$ $2^2=4$ $2^3=8$





"If there are various logical ways to explain a certain phenomenon, the simplest is the best" - Occam's Razor

Variable Selection

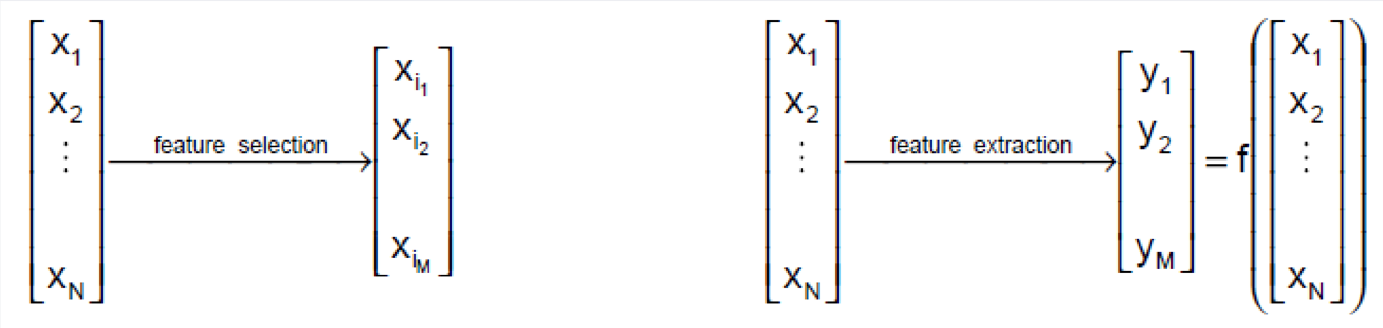
- Backgrounds
  - ✓ Theoretically, model performance improves when the number of variables increases (Under variable independence condition)
  - ✓ In reality, model performance degenerates due to variable dependence, existence of noise, etc.
- Purpose
  - ✓ Identify a subset of variables that best fit the model
- Effect
  - ✓ Remove correlations between variables
  - ✓ Simplified post-processing
  - ✓ Remove redundant or unnecessary variables while keeping relevant information
  - ✓ Visualization can be possible

Variable Selection

- Supervised vs. Unsupervised Dimensionality Reduction
  - ✓ Supervised dimensionality reduction
    - Use data mining models to verify the reduced dimensions
    - Dimensionality reduction results can be different according to the data mining algorithms employed
  - ✓ Unsupervised dimensionality reduction
    - Do not use data mining models during the process
    - Dimensionality reduction results are identical if the data and method is same

Variable Selection

- Dimensionality reduction techniques
  - ✓ Variable/feature selection
    - Select a subset of variables from the original variable set
    - Filter – Variable selection and model training are independent
    - Wrapper – Variable selection is done to optimizes the result of the considered data mining model
  - ✓ Variable/feature extraction
    - Extract a new smaller set of variables that preserve the characteristics of the original data
    - Performance metric that is independent from data mining models is used



Variable Selection

- Selection vs. Extraction
  - ✓ Conceptual difference between variable selection and variable extraction

X_1	X_2	X_3	…	X_n
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

Variable selection

X_1	X_5	X_8
…	…	…
…	…	…
…	…	…
…	…	…
…	…	…

Variable extraction

Z_1	Z_2	Z_3
…	…	…
…	…	…
…	…	…
…	…	…
…	…	…

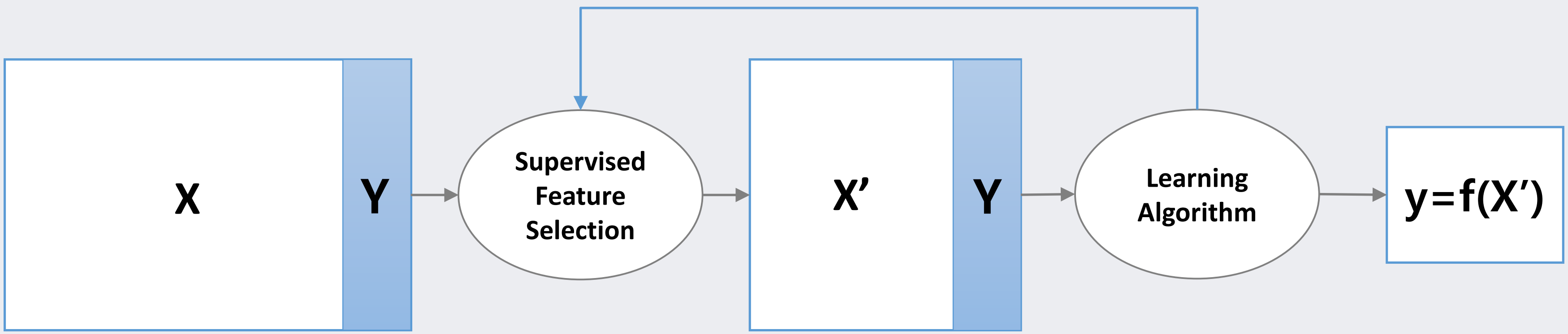
$Z_1 = X_1 + 0.2*X_2$

$Z_2 = X_3 - 2*X_5$

$Z_3 = X_4 + X_6 - X_9$

Variable Selection

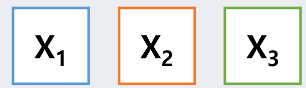
- Supervised variable selection
  - ✓ Select d’ variables from d variables (d’ << d) in order to optimize the performance of the considered learning algorithm



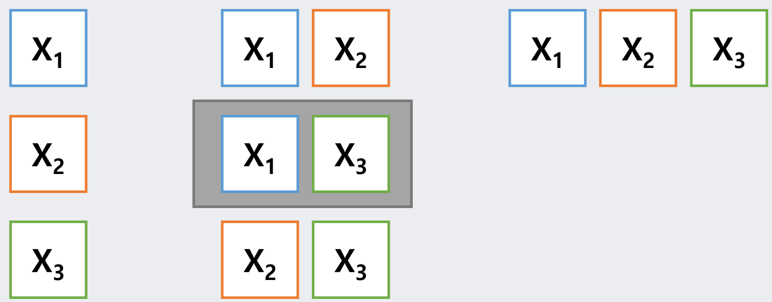
  - ✓ Select the learning algorithm before variable selection
  - ✓ Different variable selection results are possible due to the variety of learning algorithms

Variable Selection

- Exhaustive search
  - ✓ Search all possible combinations
    - Ex) 3 variables



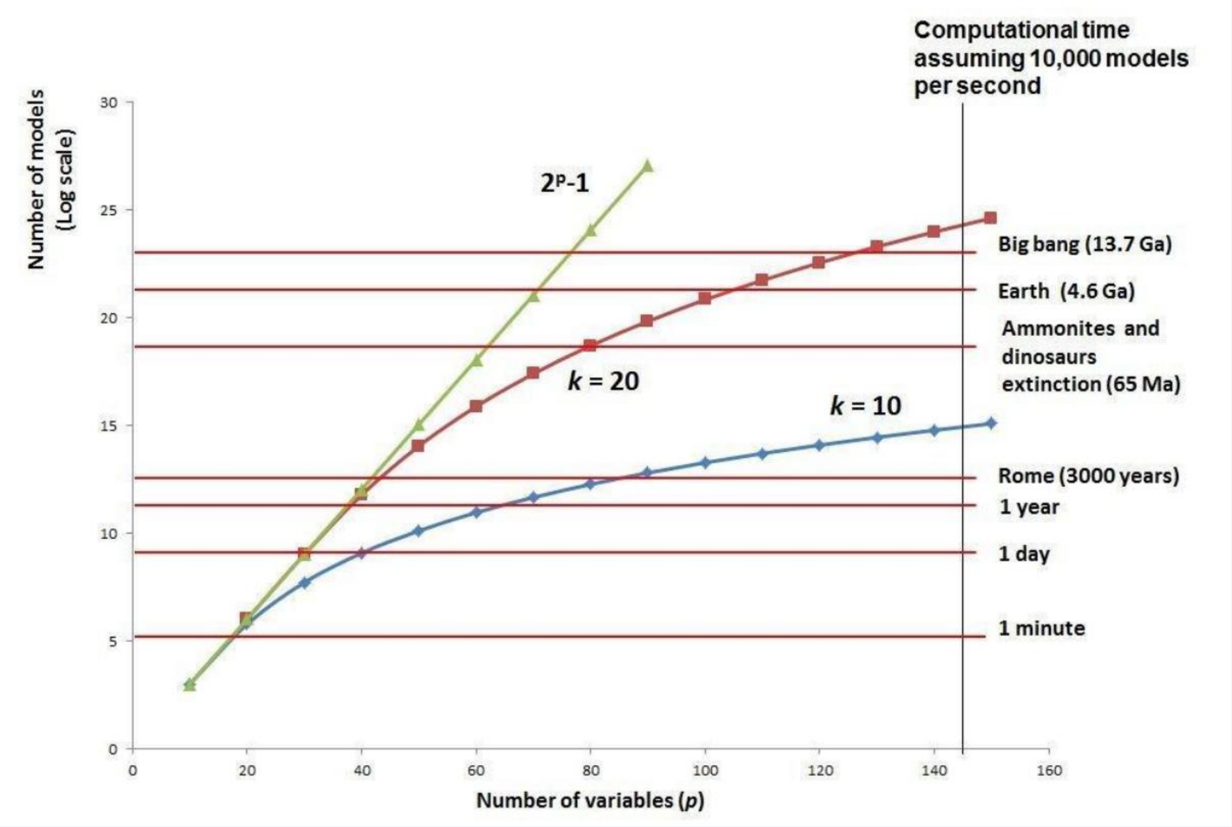
    - A total of 6 possible subsets are tested



  - ✓ Performance criteria for variable selection
    - Akaike Information Criteria (AIC), Bayesian Information Criteria (BIC), Adjusted R^2, Mallow's C_p, etc.

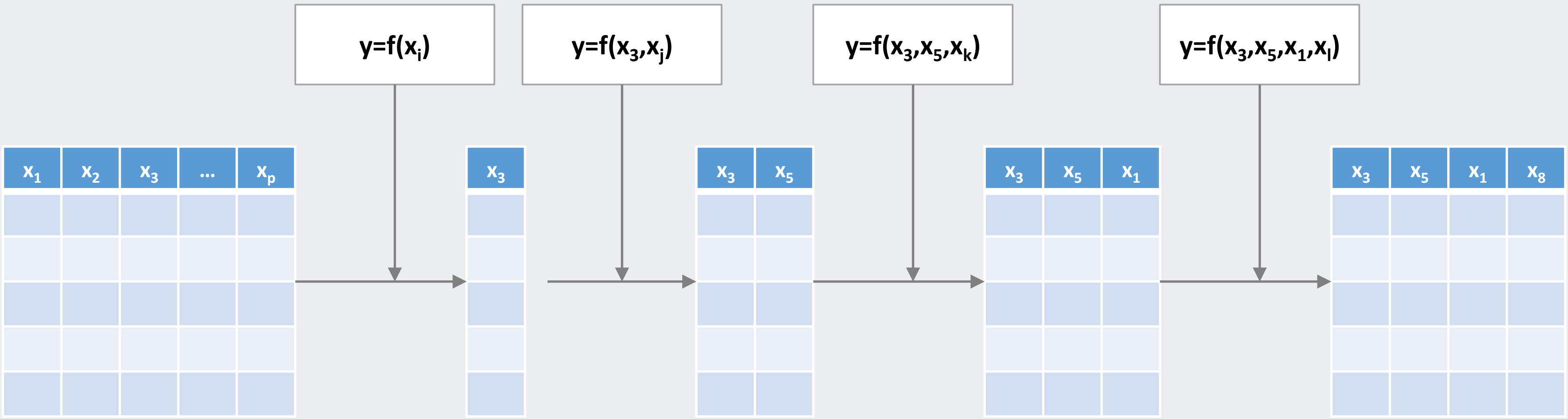
Variable Selection

- Exhaustive search
  - ✓ Assume that we have a computer that can evaluate 10,000 models/second



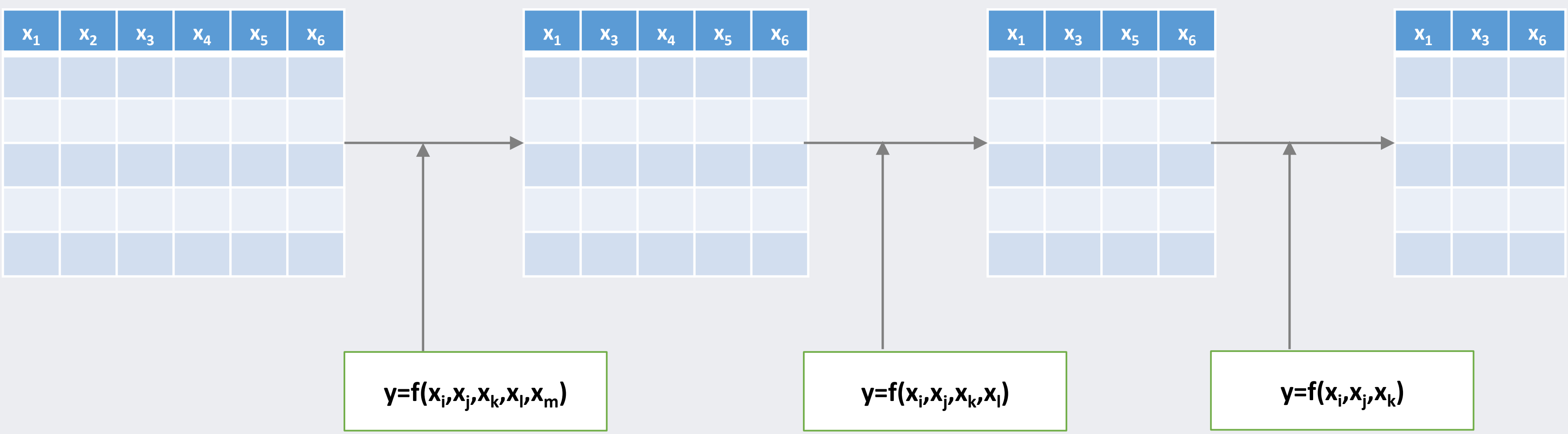
Variable Selection

- Forward selection
  - ✓ From the model with no variable, significant variables are sequentially added
  - ✓ Once the variable is selected, it will never be removed (The number of variables gradually increases)



Variable Selection

- Backward Elimination
  - ✓ From the model with all variables, irrelevant variables are sequentially removed
  - ✓ Once a variable is removed, it will never be selected (The number of variables gradually decreases)

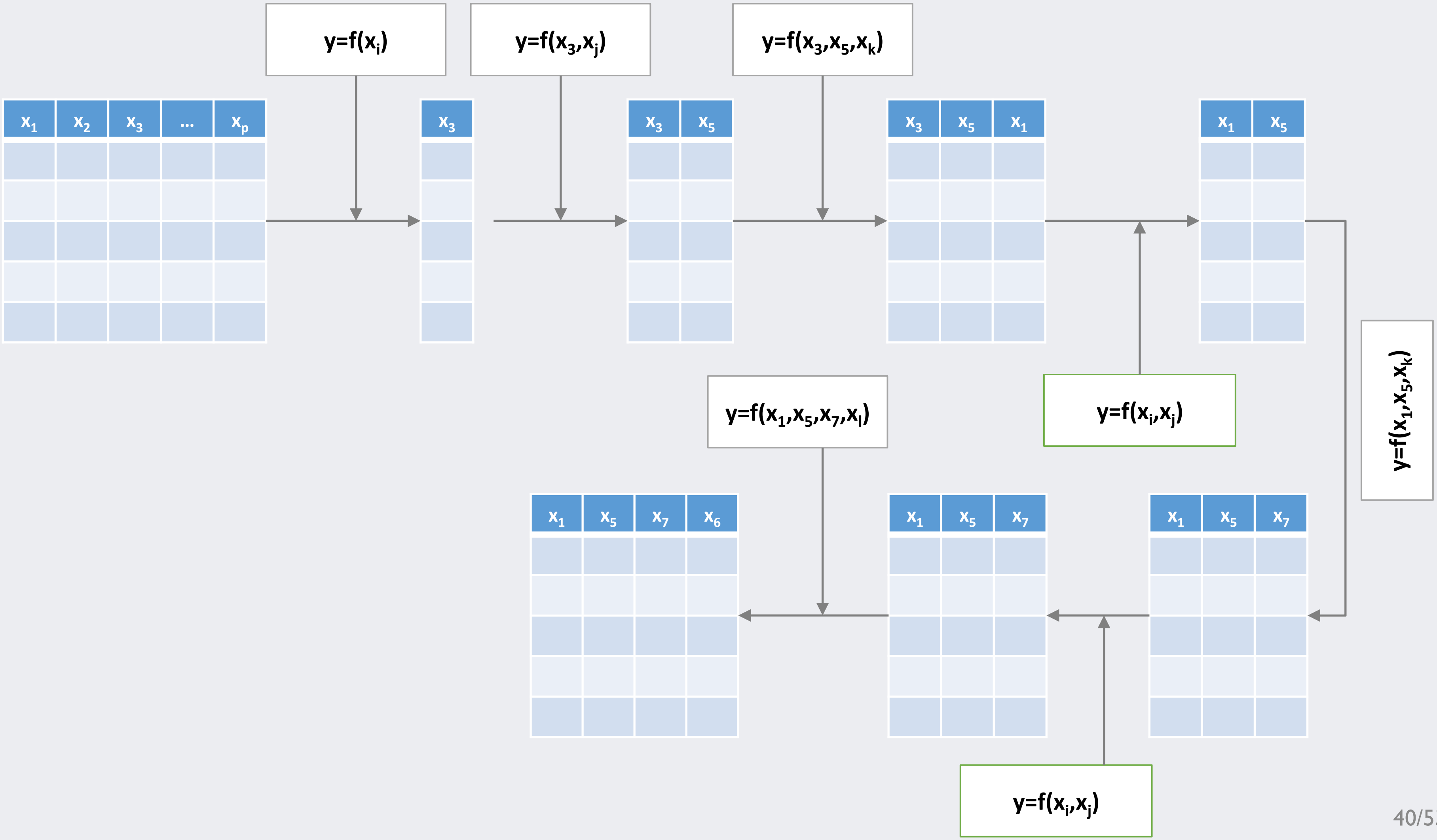


Variable Selection

- Stepwise Selection
  - ✓ From the model with no variable, conduct the forward selection and backward elimination alternately
  - ✓ Takes longer time than forward selection/backward elimination, but has more chances to find the optimal set of variables
  - ✓ Variables that is either selected/removed can be reconsidered for selection/removal
  - ✓ The number of variables increases in the early period, but it can either increase or decrease

Variable Selection

- Stepwise selection example



AGENDA

R Exercise

- Data Set: Toyota Corolla Selling Price


Variable	Description	Variable	Description
		Guarantee_Period	Guarantee period in months
		ABS	Anti-Lock Brake System (Yes=1, No=0)
Price	Offer Price in EUROs	Airbag_1	Driver_Airbag (Yes=1, No=0)
Age_08_04	Age in months as in August 2004	Airbag_2	Passenger Airbag (Yes=1, No=0)
Mfg_Month	Manufacturing month (1-12)	Airco	Airconditioning (Yes=1, No=0)
Mfg_Year	Manufacturing Year	Automatic_airco	Automatic Airconditioning (Yes=1, No=0)
KM	Accumulated Kilometers on odometer	Boardcomputer	Boardcomputer (Yes=1, No=0)
Fuel_Type	Fuel Type (Petrol, Diesel, CNG)	CD_Player	CD Player (Yes=1, No=0)
HP	Horse Power	Central_Lock	Central Lock (Yes=1, No=0)
Met_Color	Metallic Color? (Yes=1, No=0)	Powered_Windows	Powered Windows (Yes=1, No=0)
Automatic	Automatic ((Yes=1, No=0)	Power_Steering	Power Steering (Yes=1, No=0)
CC	Cylinder Volume in cubic centimeters	Radio	Radio (Yes=1, No=0)
Doors	Number of doors	Mistlamps	Mistlamps (Yes=1, No=0)
Cylinders	Number of cylinders	Sport_Model	Sport Model (Yes=1, No=0)
Gears	Number of gear positions	Backseat_Divider	Backseat Divider (Yes=1, No=0)
Quarterly_Tax	Quarterly road tax in EUROs	Metallic_Rim	Metallic Rim (Yes=1, No=0)
Weight	Weight in Kilograms	Radio_cassette	Radio Cassette (Yes=1, No=0)
Mfr_Guarantee	Within Manufacturer's Guarantee period (Yes=1, No=0)	Parking_Assistant	Parking assistance system (Yes=1, No=0)
BOVAG_Guarantee	BOVAG (Dutch dealer network) Guarantee (Yes=1, No=0)	Tow_Bar	Tow Bar (Yes=1, No=0)

R Exercise

- Import the dataset & preprocessing
  - ✓ Convert categorical variable to binary variables (1-of-c coding)

```r
# Working directory 지정
setwd("C:\\RStudy")

# 실습 1: 전진선택/후진소거/단계적선택 ----------------------
# 분석에 필요한 패키지 설치 및 불러오기
# Multivariate linear regression
corolla <- read.csv("ToyotaCorolla.csv")

# Indices for the activated input variables
nCar <- dim(corolla)[1]
nVar <- dim(corolla)[2]

id_idx <- c(1,2)
category_idx <- 8

# 범주형 변수를 이진형 변수로 변환
dummy_p <- rep(0,nCar)
dummy_d <- rep(0,nCar)
dummy_c <- rep(0,nCar)

p_idx <- which(corolla$Fuel_Type == "Petrol")
d_idx <- which(corolla$Fuel_Type == "Diesel")
c_idx <- which(corolla$Fuel_Type == "CNG")

dummy_p[p_idx] <- 1
dummy_d[d_idx] <- 1
dummy_c[c_idx] <- 1

Fuel <- data.frame(dummy_p, dummy_d, dummy_c)
names(Fuel) <- c("Petrol","Diesel","CNG")

# Prepare the data for MLR
mlr_data <- cbind(corolla[,-c(id_idx, category_idx)], Fuel)
```

Price	Age_08_04	Mfg_Month	Mfg_Year	KM	Fuel_Type	HP	Met_Color	Automatic	cc
13500	23	10	2002	46986	Diesel	90	1	0	2000
13750	23	10	2002	72937	Diesel	90	1	0	2000
13950	24	9	2002	41711	Diesel	90	1	0	2000
14950	26	7	2002	48000	Diesel	90	0	0	2000
13750	30	3	2002	38500	Diesel	90	0	0	2000
12950	32	1	2002	61000	Diesel	90	0	0	2000
16900	27	6	2002	94612	Diesel	90	1	0	2000
18600	30	3	2002	75889	Diesel	90	1	0	2000
21500	27	6	2002	19700	Petrol	192	0	0	1800
12950	23	10	2002	71138	Diesel	69	0	0	1900
20950	25	8	2002	31461	Petrol	192	0	0	1800
19950	22	11	2002	43610	Petrol	192	0	0	1800
19600	25	8	2002	32189	Petrol	192	0	0	1800
21500	31	2	2002	23000	Petrol	192	1	0	1800
22500	32	1	2002	34131	Petrol	192	1	0	1800

KM	HP	Met_Color
46986	90	1
72937	90	1
41711	90	1
48000	90	0
38500	90	0
61000	90	0
94612	90	1
75889	90	1
19700	192	0
71138	69	0
31461	192	0
43610	192	0
32189	192	0
23000	192	1

…

Petrol	Diesel	CNG
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
1	0	0
0	1	0
1	0	0
1	0	0
1	0	0
1	0	0

R Exercise

- Divide the dataset into training/validation

```
# Split the data into the training/validation sets
trn_idx <- sample(1:nCar, round(0.7*nCar))
trn_data <- mlr_data[trn_idx,]
val_data <- mlr_data[-trn_idx,]
```

mlr_data	1436 obs. of 37 variables
trn_data	1005 obs. of 37 variables
val_data	431 obs. of 37 variables

- MLR with all variables

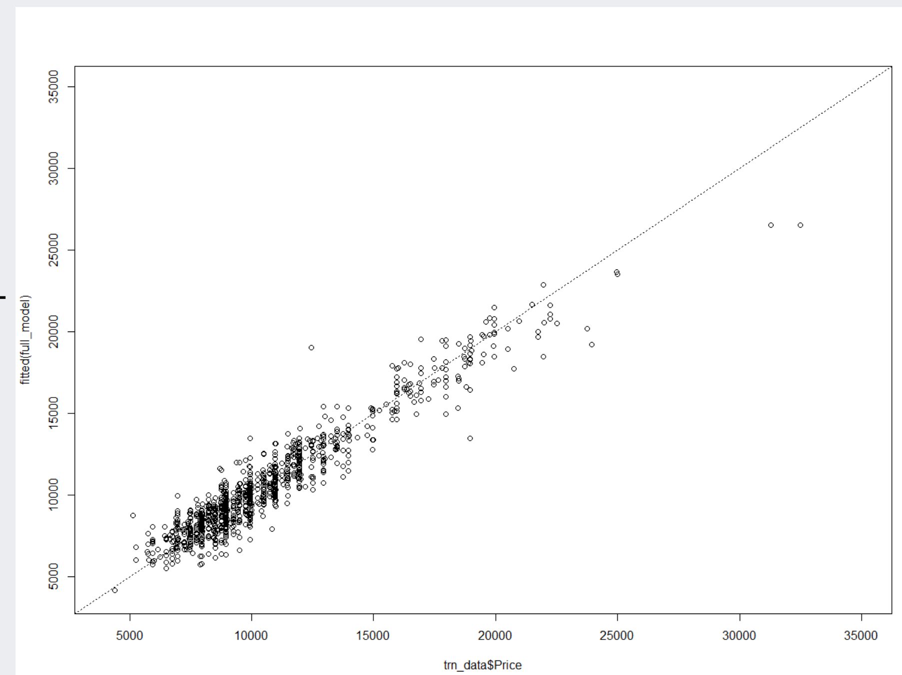
```
# Train the MLR
full_model <- lm(Price ~ ., data = trn_data)
full_model
summary(full_model)
plot(full_model)

# Plot the result
plot(trn_data$Price, fitted(full_model), xlim = c(4000,35000), ylim = c(4000,35000))
abline(0,1,lty=3)

anova(full_model)
plot(fitted(full_model), resid(full_model), xlab="Fitted values", ylab="Residuals")
```

R Exercise

- MLR with all variables

```
> summary(full_model)

Call:
lm(formula = Price ~ ., data = trn_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-6571.9  -640.9   -49.0   624.2  5972.3 

Coefficients: (3 not defined because of singularities)
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)       3.540e+03  1.724e+03   2.054 0.040257 *  
Age_08_04        -1.178e+02  3.914e+00 -30.104  < 2e-16 ***
Mfg_Month        -1.059e+02  1.034e+01 -10.244  < 2e-16 ***
Mfg_Year                 NA         NA      NA       NA    
KM               -1.710e-02  1.338e-03 -12.777  < 2e-16 ***
HP                1.911e+01  3.601e+00   5.305 1.39e-07 ***
Met_Color        -4.358e+01  7.632e+01  -0.571 0.568134    
Automatic         3.746e+02  1.458e+02   2.568 0.010368 *  
cc               -5.613e-02  7.515e-02  -0.747 0.455279    
Doors             7.198e+01  4.111e+01   1.751 0.080257 .  
Cylinders                NA         NA      NA       NA    
Gears             1.959e+02  2.142e+02   0.915 0.360617    
Quarterly_Tax     1.159e+01  2.128e+00   5.446 6.52e-08 ***
Weight            8.879e+00  1.227e+00   7.233 9.54e-13 ***
Mfr_Guarantee     2.360e+02  7.381e+01   3.198 0.001430 ** 
BOVAG_Guarantee   3.989e+02  1.316e+02   3.033 0.002490 ** 
Guarantee_Period  7.207e+01  1.459e+01   4.938 9.27e-07 ***
ABS              -4.715e+01  1.300e+02  -0.363 0.716844    
Airbag_1          4.498e+02  2.570e+02   1.750 0.080375 .  
Airbag_2         -2.007e+02  1.314e+02  -1.527 0.127121    
Airco             2.245e+02  8.919e+01   2.517 0.012008 *  
Automatic_airco   2.435e+03  1.889e+02  12.890  < 2e-16 ***
Boardcomputer    -2.099e+02  1.194e+02  -1.758 0.078992 .  
CD_Player         8.442e+01  1.010e+02   0.836 0.403239    
Central_Lock     -7.471e+01  1.419e+02  -0.526 0.598678    
Powered_Windows   5.112e+02  1.424e+02   3.589 0.000349 ***
Power_Steering   -5.689e+02  2.842e+02  -2.002 0.045581 *  
Radio             5.575e+02  6.295e+02   0.886 0.376037    
Mistlamps         1.869e+01  1.102e+02   0.170 0.865286    
Sport_Model       2.790e+02  8.906e+01   3.132 0.001787 ** 
Backseat_Divider -6.961e+01  1.327e+02  -0.525 0.599953    
Metallic_Rim      5.536e+01  9.675e+01   0.572 0.567342    
Radio_cassette   -5.593e+02  6.299e+02  -0.888 0.374863    
Tow_Bar          -1.990e+02  8.018e+01  -2.482 0.013216 *  
Petrol            1.096e+03  4.339e+02   2.527 0.011663 *  
Diesel            5.269e+02  4.128e+02   1.276 0.202180    
CNG                      NA         NA      NA       NA    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1060 on 971 degrees of freedom
Multiple R-squared:  0.9127,	Adjusted R-squared:  0.9098 
F-statistic: 307.7 on 33 and 971 DF,  p-value: < 2.2e-16
```

Estimated price



Actual price

R Exercise

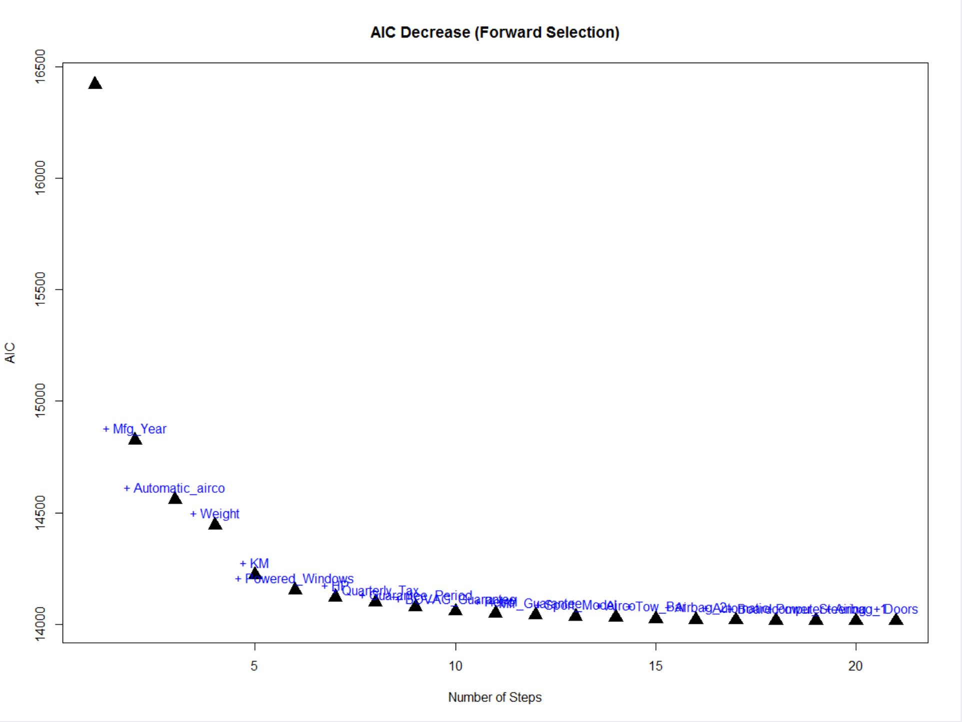
- Variable selection 1: Forward selection
  - ✓ Starts with zero variable and adds the most significant variable at once

```r
# 변수선택 1: 전진선택법
# Upperbound formula 만들기
tmp_x <- paste(colnames(trn_data)[-1], collapse=" + ")
tmp_xy <- paste("Price ~ ", tmp_x, collapse = "")
tmp_xy
as.formula(tmp_xy)

forward_model <- step(lm(Price ~ 1, data = trn_data),
                      scope = list(upper = as.formula(tmp_xy), lower = Price ~ 1), direction="forward", trace=1)
summary(forward_model)
anova(forward_model)

# 각 단계에서 선택된 변수 표시
forward_model$anova$Step
forward_model$anova$AIC

# 선택된 변수에 따른 AIC 감소분 표시
plot(forward_model$anova$AIC, pch = 17, cex=2, main = "AIC Decrease (Forward Selection)", xlab = "Number of Steps", ylab = "AIC")
text(forward_model$anova$AIC, forward_model$anova$Step, cex=1, pos=3, col="blue")
```

R Exercise

- Variable selection 1: Forward selection
  - ✓ Variable selection results (36 variables → 20 variables)

```
> summary(forward_model)

Call:
lm(formula = Price ~ Mfg_Year + Automatic_airco + Weight + KM +
    Powered_Windows + HP + Quarterly_Tax + Guarantee_Period +
    BOVAG_Guarantee + Petrol + Mfr_Guarantee + Sport_Model +
    Airco + Tow_Bar + Airbag_2 + Automatic + Boardcomputer +
    Power_Steering + Airbag_1 + Doors, data = trn_data)

Residuals:
    Min      1Q  Median      3Q     Max
-6747.2  -653.8   -53.8   640.8  5908.7

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)      -2.807e+06  8.542e+04 -32.857  < 2e-16 ***
Mfg_Year          1.402e+03  4.286e+01  32.718  < 2e-16 ***
Automatic_airco   2.451e+03  1.746e+02  14.037  < 2e-16 ***
Weight            9.233e+00  1.166e+00   7.918 6.50e-15 ***
KM               -1.734e-02  1.309e-03 -13.252  < 2e-16 ***
Powered_Windows   4.650e+02  8.300e+01   5.602 2.74e-08 ***
HP                1.819e+01  3.235e+00   5.625 2.42e-08 ***
Quarterly_Tax     1.146e+01  2.013e+00   5.694 1.63e-08 ***
Guarantee_Period  7.624e+01  1.377e+01   5.535 3.98e-08 ***
BOVAG_Guarantee   4.078e+02  1.268e+02   3.216 0.001342 **
Petrol            6.593e+02  2.956e+02   2.231 0.025933 *
Mfr_Guarantee     2.263e+02  7.214e+01   3.137 0.001757 **
Sport_Model       2.811e+02  8.226e+01   3.417 0.000659 ***
Airco             2.430e+02  8.500e+01   2.859 0.004334 **
Tow_Bar          -2.203e+02  7.742e+01  -2.846 0.004523 **
Airbag_2         -2.167e+02  9.707e+01  -2.232 0.025847 *
Automatic         3.395e+02  1.428e+02   2.378 0.017586 *
Boardcomputer    -1.929e+02  1.126e+02  -1.713 0.087054 .
Power_Steering   -6.486e+02  2.715e+02  -2.389 0.017075 *
Airbag_1          4.773e+02  2.521e+02   1.893 0.058598 .
Doors             5.867e+01  3.958e+01   1.482 0.138578
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1056 on 984 degrees of freedom
Multiple R-squared:  0.9121, Adjusted R-squared:  0.9103
F-statistic: 510.6 on 20 and 984 DF,  p-value: < 2.2e-16
```



R Exercise

- Variable selection 2: Backward elimination
  - ✓ Starts with all variable and removes the most insignificant variable at once

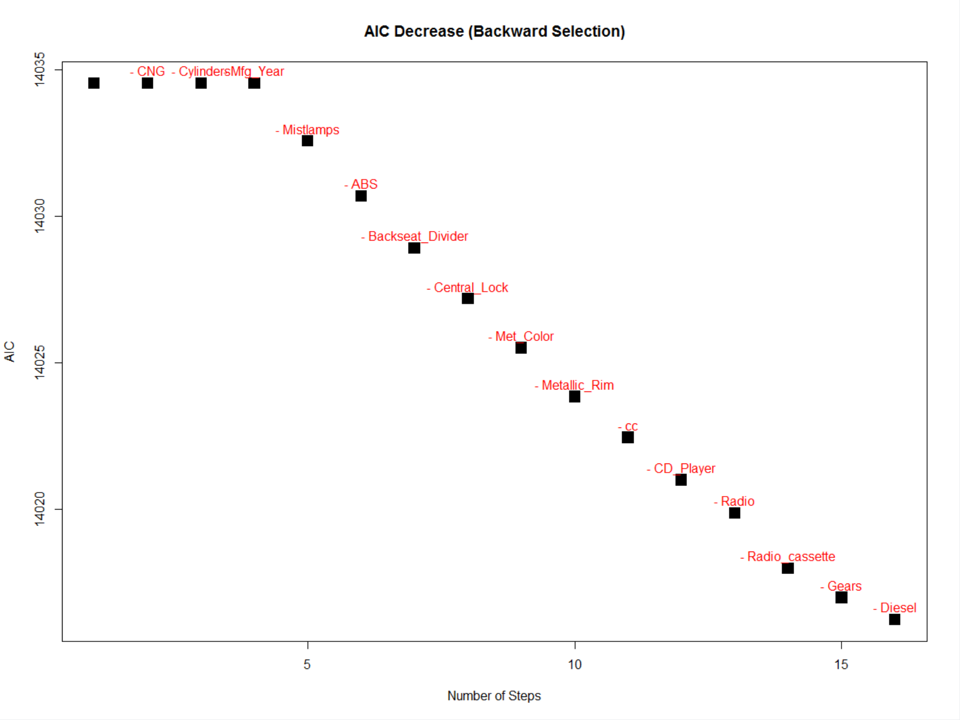
```r
# 변수선택 2: 후진소거법
backward_model <- step(full_model, scope = list(upper = as.formula(tmp_xy), lower = Price ~ 1), direction="backward", trace=1)
summary(backward_model)
anova(backward_model)

# 각 단계에서 제거된 변수 표시
backward_model$anova$Step

# 제거된 변수에 따른 AIC 감소분 표시
plot(backward_model$anova$AIC, pch = 15, cex=2, main = "AIC Decrease (Backward Selection)", xlab = "Number of Steps", ylab = "AIC")
text(backward_model$anova$AIC, backward_model$anova$Step, cex=1, pos=3, col="red")
```

```
> backward_model$anova$Step
 [1] ""                  "- CNG"             "- Cylinders"       "- Mfg_Year"        "- Mistlamps"
 [6] "- ABS"             "- Backseat_Divider" "- Central_Lock"   "- Met_Color"       "- Metallic_Rim"
[11] "- cc"              "- CD_Player"       "- Radio"           "- Radio_cassette"  "- Gears"
[16] "- Diesel"
```

R Exercise

- Variable selection 2: Backward elimination
  - ✓ Variable selection results (36 variables → 21 variables)

```
> summary(backward_model)

Call:
lm(formula = Price ~ Age_08_04 + Mfg_Month + KM + HP + Automatic + 
    Doors + Quarterly_Tax + Weight + Mfr_Guarantee + BOVAG_Guarantee + 
    Guarantee_Period + Airbag_1 + Airbag_2 + Airco + Automatic_airco + 
    Boardcomputer + Powered_Windows + Power_Steering + Sport_Model + 
    Tow_Bar + Petrol, data = trn_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-6744.2  -643.7   -43.5   630.5  5924.2 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)       4.528e+03  1.309e+03   3.460 0.000563 ***
Age_08_04        -1.176e+02  3.644e+00 -32.281  < 2e-16 ***
Mfg_Month        -1.067e+02  1.021e+01 -10.456  < 2e-16 ***
KM               -1.711e-02  1.327e-03 -12.898  < 2e-16 ***
HP                1.809e+01  3.236e+00   5.590 2.95e-08 ***
Automatic         3.373e+02  1.428e+02   2.363 0.018341 *  
Doors             5.612e+01  3.965e+01   1.415 0.157290    
Quarterly_Tax     1.156e+01  2.015e+00   5.738 1.27e-08 ***
Weight            9.259e+00  1.166e+00   7.938 5.57e-15 ***
Mfr_Guarantee     2.249e+02  7.215e+01   3.117 0.001879 ** 
BOVAG_Guarantee   4.138e+02  1.269e+02   3.260 0.001150 ** 
Guarantee_Period  7.511e+01  1.381e+01   5.437 6.82e-08 ***
Airbag_1          4.597e+02  2.526e+02   1.820 0.069067 .  
Airbag_2         -2.272e+02  9.758e+01  -2.329 0.020071 *  
Airco             2.377e+02  8.514e+01   2.791 0.005351 ** 
Automatic_airco   2.455e+03  1.746e+02  14.060  < 2e-16 ***
Boardcomputer    -2.056e+02  1.133e+02  -1.816 0.069700 .  
Powered_Windows   4.620e+02  8.304e+01   5.563 3.41e-08 ***
Power_Steering   -6.192e+02  2.729e+02  -2.269 0.023479 *  
Sport_Model       2.717e+02  8.273e+01   3.284 0.001059 ** 
Tow_Bar          -2.156e+02  7.754e+01  -2.780 0.005531 ** 
Petrol            7.055e+02  2.988e+02   2.361 0.018404 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1056 on 983 degrees of freedom
Multiple R-squared:  0.9122,	Adjusted R-squared:  0.9103 
F-statistic: 486.4 on 21 and 983 DF,  p-value: < 2.2e-16
```



R Exercise

- Variable selection 3: Stepwise selection
  - ✓ Starts with zero variable and alternately adds the most significant variable and removes the most insignificant variable

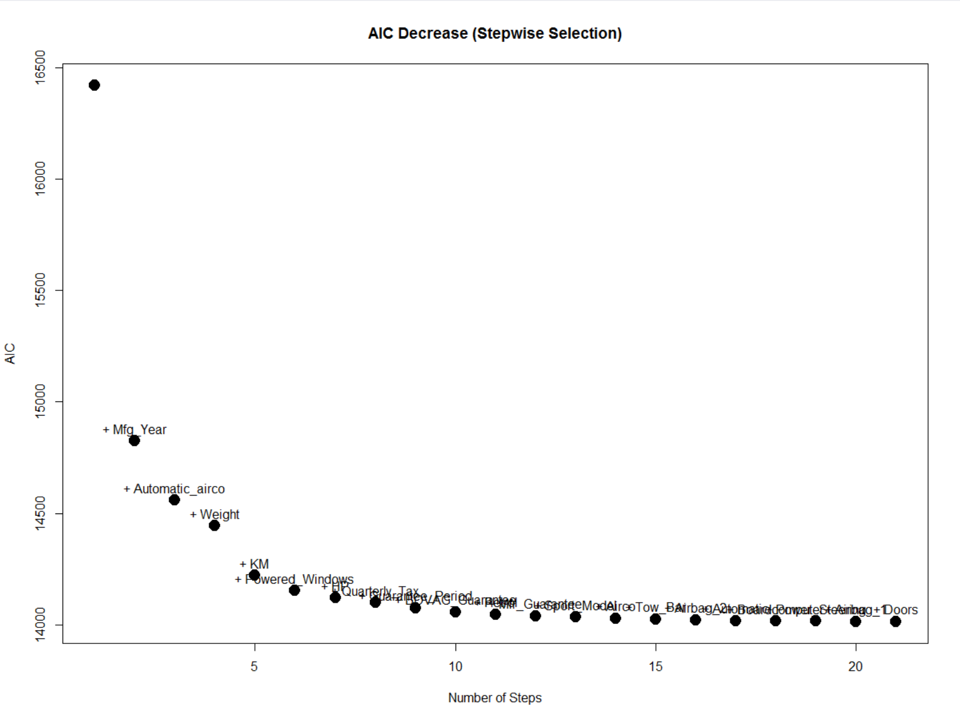
```r
# 변수선택 3: 단계적 선택법
stepwise_model <- step(lm(Price ~ 1, data = trn_data),
                       scope = list(upper = as.formula(tmp_xy), lower = Price ~ 1), direction="both", trace=1)
summary(stepwise_model)
anova(stepwise_model)

# 각 단계에서 선택/제거된 변수 표시
stepwise_model$anova$Step
stepwise_model$anova$AIC

# 제거/선택된 변수에 따른 AIC 감소분 표시
plot(stepwise_model$anova$AIC, pch = 19, cex=2, main = "AIC Decrease (Stepwise Selection)", xlab = "Number of Steps", ylab = "AIC")
text(stepwise_model$anova$AIC, stepwise_model$anova$Step, cex=1, pos=3, col="black")
```

```
> stepwise_model$anova$Step
 [1] ""                    "+ Mfg_Year"          "+ Automatic_airco"  "+ Weight"            "+ KM"                "+ Powered_Windows"
 [7] "+ HP"                "+ Quarterly_Tax"     "+ Guarantee_Period" "+ BOVAG_Guarantee"   "+ Petrol"            "+ Mfr_Guarantee"
[13] "+ Sport_Model"       "+ Airco"             "+ Tow_Bar"          "+ Airbag_2"          "+ Automatic"         "+ Boardcomputer"
[19] "+ Power_Steering"    "+ Airbag_1"          "+ Doors"
```

R Exercise

- Variable selection 3: Stepwise selection
  - ✓ Variable selection result

```
> summary(stepwise_model)

Call:
lm(formula = Price ~ Mfg_Year + Automatic_airco + Weight + KM +
    Powered_Windows + HP + Quarterly_Tax + Guarantee_Period +
    BOVAG_Guarantee + Petrol + Mfr_Guarantee + Sport_Model +
    Airco + Tow_Bar + Airbag_2 + Automatic + Boardcomputer +
    Power_Steering + Airbag_1 + Doors, data = trn_data)

Residuals:
    Min      1Q  Median      3Q     Max
-6747.2  -653.8   -53.8   640.8  5908.7

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)      -2.807e+06  8.542e+04 -32.857  < 2e-16 ***
Mfg_Year          1.402e+03  4.286e+01  32.718  < 2e-16 ***
Automatic_airco   2.451e+03  1.746e+02  14.037  < 2e-16 ***
Weight            9.233e+00  1.166e+00   7.918 6.50e-15 ***
KM               -1.734e-02  1.309e-03 -13.252  < 2e-16 ***
Powered_Windows   4.650e+02  8.300e+01   5.602 2.74e-08 ***
HP                1.819e+01  3.235e+00   5.625 2.42e-08 ***
Quarterly_Tax     1.146e+01  2.013e+00   5.694 1.63e-08 ***
Guarantee_Period  7.624e+01  1.377e+01   5.535 3.98e-08 ***
BOVAG_Guarantee   4.078e+02  1.268e+02   3.216 0.001342 **
Petrol            6.593e+02  2.956e+02   2.231 0.025933 *
Mfr_Guarantee     2.263e+02  7.214e+01   3.137 0.001757 **
Sport_Model       2.811e+02  8.226e+01   3.417 0.000659 ***
Airco             2.430e+02  8.500e+01   2.859 0.004334 **
Tow_Bar          -2.203e+02  7.742e+01  -2.846 0.004523 **
Airbag_2         -2.167e+02  9.707e+01  -2.232 0.025847 *
Automatic         3.395e+02  1.428e+02   2.378 0.017586 *
Boardcomputer    -1.929e+02  1.126e+02  -1.713 0.087054 .
Power_Steering   -6.486e+02  2.715e+02  -2.389 0.017075 *
Airbag_1          4.773e+02  2.521e+02   1.893 0.058598 .
Doors             5.867e+01  3.958e+01   1.482 0.138578
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1056 on 984 degrees of freedom
Multiple R-squared:  0.9121, Adjusted R-squared:  0.9103
F-statistic: 510.6 on 20 and 984 DF,  p-value: < 2.2e-16
```



R Exercise

- Performance comparison among the variable selection techniques

```r
# 검증 데이터에 대한 각 변수선택 결과의 예측 정확도 비교
full_haty <- predict(full_model, newdata = val_data)
forward_haty <- predict(forward_model, newdata = val_data)
backward_haty <- predict(backward_model, newdata = val_data)
stepwise_haty <- predict(stepwise_model, newdata = val_data)

# 회귀분석 예측성능 평가지표
# 1: Mean squared error (MSE)
perf_mat <- matrix(0,4,6)
perf_mat[1,1] <- mean((val_data$Price-full_haty)^2)
perf_mat[1,2] <- mean((val_data$Price-forward_haty)^2)
perf_mat[1,3] <- mean((val_data$Price-backward_haty)^2)
perf_mat[1,4] <- mean((val_data$Price-stepwise_haty)^2)

# 2: Root mean squared error (RMSE)
perf_mat[2,1] <- sqrt(mean((val_data$Price-full_haty)^2))
perf_mat[2,2] <- sqrt(mean((val_data$Price-forward_haty)^2))
perf_mat[2,3] <- sqrt(mean((val_data$Price-backward_haty)^2))
perf_mat[2,4] <- sqrt(mean((val_data$Price-stepwise_haty)^2))

# 3: Mean absolute error (MAE)
perf_mat[3,1] <- mean(abs(val_data$Price-full_haty))
perf_mat[3,2] <- mean(abs(val_data$Price-forward_haty))
perf_mat[3,3] <- mean(abs(val_data$Price-backward_haty))
perf_mat[3,4] <- mean(abs(val_data$Price-stepwise_haty))

# 4: Mean absolute percentage error (MAPE)
perf_mat[4,1] <- mean(abs((val_data$Price-full_haty)/val_data$Price))*100
perf_mat[4,2] <- mean(abs((val_data$Price-forward_haty)/val_data$Price))*100
perf_mat[4,3] <- mean(abs((val_data$Price-backward_haty)/val_data$Price))*100
perf_mat[4,4] <- mean(abs((val_data$Price-stepwise_haty)/val_data$Price))*100

# 변수선택 기법 결과 비교
rownames(perf_mat) <- c("MSE", "RMSE", "MAE", "MAPE")
colnames(perf_mat) <- c("All", "Forward", "Backward", "Stepwise", "GA_default", "GA_yourOwn")
perf_mat
```

```
> perf_mat
              All      Forward     Backward     Stepwise G
MSE  1.577365e+06 1.634343e+06 1.623485e+06 1.634343e+06
RMSE 1.255932e+03 1.278414e+03 1.274160e+03 1.278414e+03
MAE  9.121387e+02 9.242534e+02 9.211011e+02 9.242534e+02
MAPE 9.428209e+00 9.538071e+00 9.497384e+00 9.538071e+00
```