

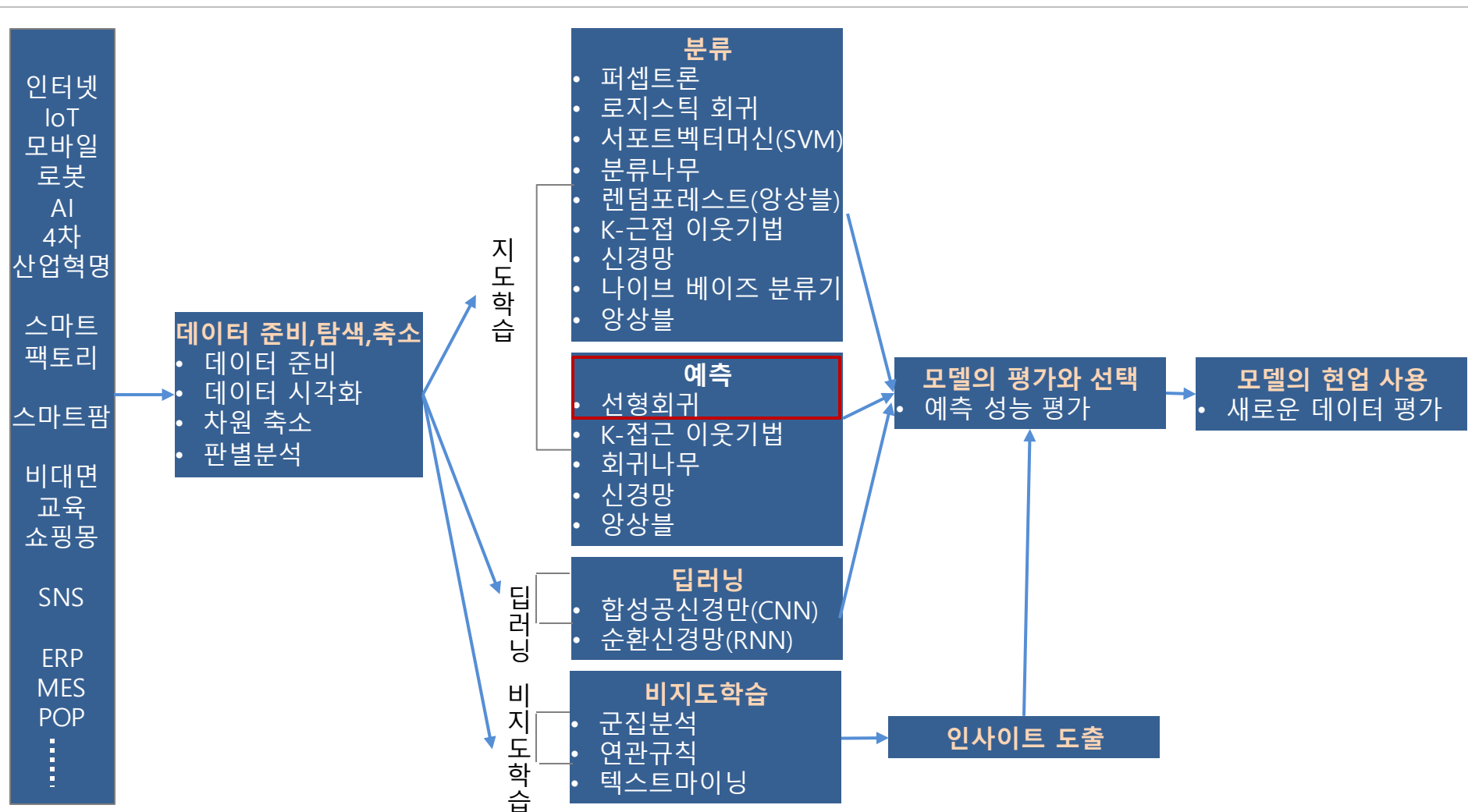


9주차: 모델링과 예측 : 선형회귀

ChulSoo Park

School of Computer Engineering & Information Technology
Korea National University of Transportation

■ 데이터 분석 Process에서 이번주 교육 위치



7.3 단순 선형 회귀

- 앞 절에서 설명한 것과 같이 선형 회귀에서는 최적화 문제를 풀어야 함
 - 최적화는 미분을 이용하여 해결함
 - R은 이 문제를 푸는 `lm(linear models)`이라는 함수를 제공

The screenshot displays the RStudio interface. The left pane shows the source editor with the following R code:

```
12 plot(cars, type="s", main = "cars")
13
14 library(gapminder)
15 library(dplyr)
16 library(ggplot2)
17 library(RColorBrewer)
18
19 ?lm
20
21
22
23
```

The bottom pane shows the console with the prompt `>`.

The right pane shows the 'Environment' tab selected, with the 'Files' sub-tab active. It displays the documentation for the `lm` function from the `stats` package, titled 'Fitting Linear Models'. The documentation includes a description, usage, and arguments.

Description

`lm` is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although [aov](#) may provide a more convenient interface for these).

Usage

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Arguments

7.3 단순 선형 회귀

■ lm을 이용한 모델 적합 예

- 2절의 데이터 예제를 재사용

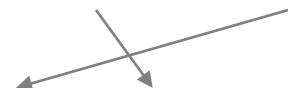
	1차	2차	3차	4차
X	3.0	6.0	9.0	12.0
Y	3.0	4.0	5.5	6.5

```
Console C:/RSources/ [Workspace loaded from C:/RSources/.RData]

> x=c(3.0,6.0,9.0,12.0) # 설명 변수 x
> y=c(3.0,4.0,5.5,6.5) # 반응 변수 y
> m=lm(y ~ x)          # 모델 적합(학습)
> m

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)          x 
        1.75         0.40
```


$$y=0.4x+1.75$$

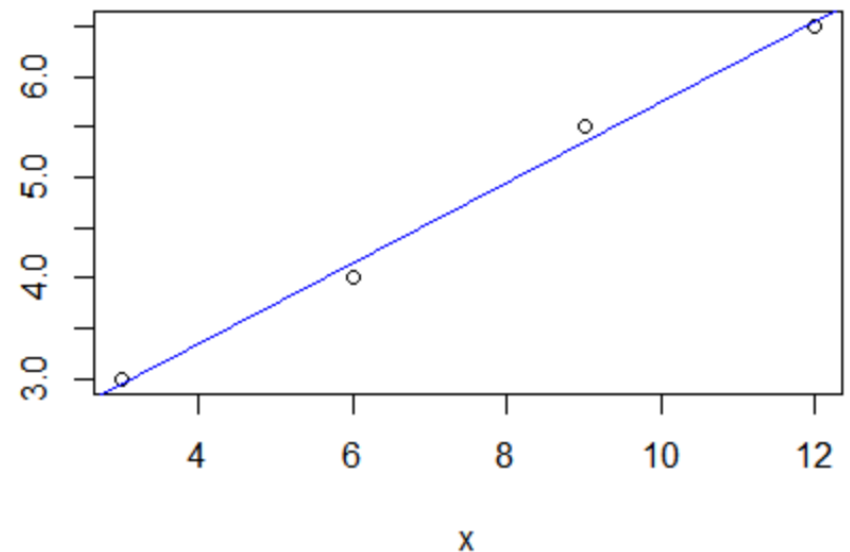
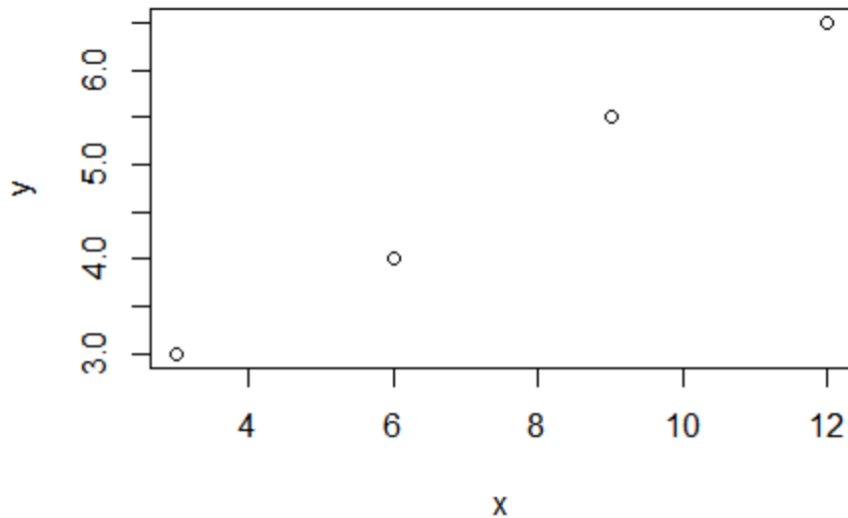
7.3 단순 선형 회귀

■ lm을 이용한 모델 적합 예

- lm 명령어는 최적 모델 m을 찾음
- m: $y=0.4x+1.75$
- 모델 m을 그림으로 그려보면

Console C:/RSources/ ↗

```
> library(dplyr)  dplyr : filter, select 와 같은 함수를 사용해 데이터 가공  
> library(ggplot2)  
> library(RColorBrewer)  
> plot(x,y)  
> abline(m, col='blue')  abline 함수 : plot으로 그린 그래프 위에 모델 m을 덧씌워주는 함수
```



7.3 단순 선형 회귀

■ 데이터의 오차 분석

- 모델 $y=0.4x+1.75$ 의 오차 분석

x1	3.0	6.0	9.0	12.0
예측값	2.95	4.15	5.35	6.55
그라운드 투루스	3.0	4.0	5.5	6.5
오차	0.05	-0.15	0.15	-0.05

- 평균 제곱 오차 MSE는

$$E = \frac{1}{4} ((0.05)^2 + (-0.15)^2 + (0.15)^2 + (-0.05)^2) = 0.0125$$

7.3 단순 선형 회귀

■ 모델의 특성 살펴보기

```




Console C:/RSources/
> coef(m)           # 매개 변수(계수) 값을 알려줌
(Intercept)         x
      1.75         0.40
> fitted(m)         # 훈련 집합에 있는 샘플에 대한 예측값
      1      2      3      4
2.95 4.15 5.35 6.55
> residuals(m)      # 잔차를 알려줌
      1      2      3      4
0.05 -0.15  0.15 -0.05
> deviance(m)/length(x) # 잔차 제곱합을 평균 제곱 오차로 계산
[1] 0.0125

```

- 잔차 제곱합
$$D = \sum_{i=1}^n (y_i - f(x_i))^2 \quad \text{식 (4)}$$
- $E = D / n : (E = 1/4 ((0.05)^2 + (-0.15)^2 + (0.15)^2 + (-0.05)^2) = 0.0125)$

7.3 단순 선형 회귀

■ summary 함수로 모델을 상세하게 살피기(1)

Console C:/RSources/   

```
> summary(m)
```

call:
lm(formula = y ~ x)

Residuals:				
1	2	3	4	
0.05	-0.15	0.15	-0.05	

Training set에 대한 잔차(차이)

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	1.75000	0.19365	9.037	0.01202	*
x	0.40000	0.02357	16.971	0.00345	**

매개 변수 T-값 P-값

signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1581 on 2 degrees of freedom
 Multiple R-squared: 0.9931, Adjusted R-squared: 0.9897
 F-statistic: 288 on 1 and 2 DF, p-value: 0.003454

7.3 단순 선형 회귀

■ summary 함수로 모델을 상세하게 살피기(2)

Residuals:			
1	2	3	4
0.05	-0.15	0.15	-0.05

Training set에 대한 잔차(차이)

Coefficients:				
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.75000	0.19365	9.037	0.01202 *
x	0.40000	0.02357	16.971	0.00345 **

매개 변수 T-값 P-값

유의수준 0.05로 설정한다면, p-값이 유의수준보다 작으므로

귀무가설 ($\alpha_0=0$, 즉 독립변수 x는 종속변수 y는 아무 관련이 없다)은 기각된다.

즉 두 변수는 관련이 있다라는 대립 가설이 받아들여진다.

여기서 m은 대형 할인점이 입점한 월이며 ICI는 분석 대상 월(m)에 그룹 C에 속하는 대리점의 총 수이다. <표 6>의 그룹 A에 속하는 행들은 가설 1에 대한 각 대리점들의 유의확률(p-value)을 나타낸 것이다. 이 유의확률이 유의수준 0.05보다 작은 경우의 수는 전체 27개 표본 중에서 26개로써 96.3%의 높은 비중을 차지한다. 이는 다시 말하면 대형할인점의 입

<표 6> 그룹 A 및 그룹 B에 속하는 대리점의 매출액 변동에 대한 가설 검정 결과
(값: 가설 1 및 가설 2에 대한 유의확률)

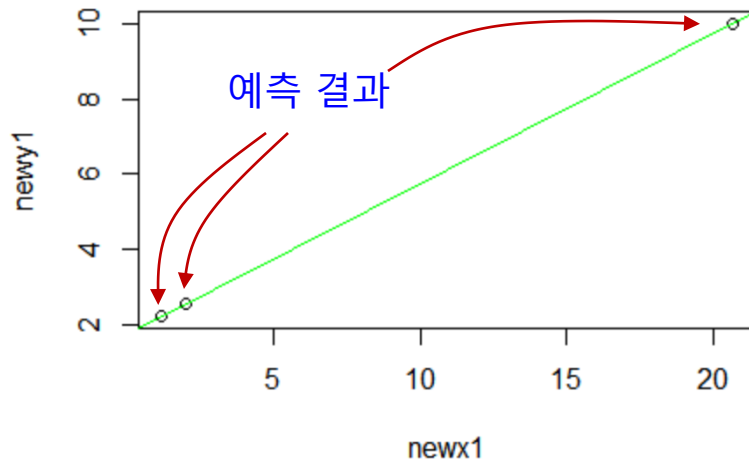
그룹	표본	1월	2월	3월	4월	5월	6월	7월	8월	9월	11월	12월
A	1	<0,000	<0,000	<0,000	<0,000	<0,000	<0,000	<0,000	0.998	<0,000	<0,000	<0,000
	2	<0,000	<0,000			<0,000		<0,000	<0,000	<0,000	<0,000	<0,000
	3		<0,000			<0,000				<0,000	<0,000	<0,000
	4					<0,000					<0,000	<0,000
B1	1	<0,000	<0,000	<0,000		0.264	<0,000	<0,000	0.002	<0,000	0.998	1,000
	2		<0,000	<0,000		<0,000	<0,000			<0,000	0.019	0.008
	3		<0,000			<0,000	<0,000			0.999	<0,000	<0,000
	4		<0,000			<0,000				<0,000	<0,000	<0,000
	5					<0,000				<0,000	<0,000	<0,000
	6					<0,000				<0,000	<0,000	
	7					<0,000					<0,000	

7.3 단순 선형 회귀

- 예측(모델을 가졌으니 예측이 가능)
 - predict 함수로 예측
 - m: $y=0.4x+1.75$
 - 예) 1.2, 2.0, 20.65라는 3개의 샘플이 새로 발생했다고 가정

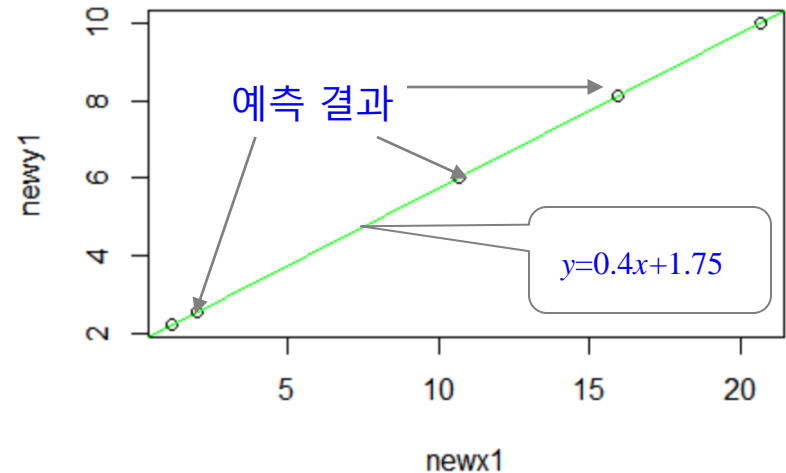
Console C:/RSources/ ↗

```
> newx =data.frame(x=c(1.2,2.0,20.65))
> predict(m, newdata=newx)
      1      2      3
2.23  2.55 10.01
```



Console C:/RSources/ ↗

```
> newx =data.frame(x=c(1.2,2.0,15.9,10.7,20.65))
> predict(m, newdata=newx)
      1      2      3      4      5
2.23  2.55  8.11  6.03 10.01
```



7.4 단순 선형 회귀의 적용 : cars data

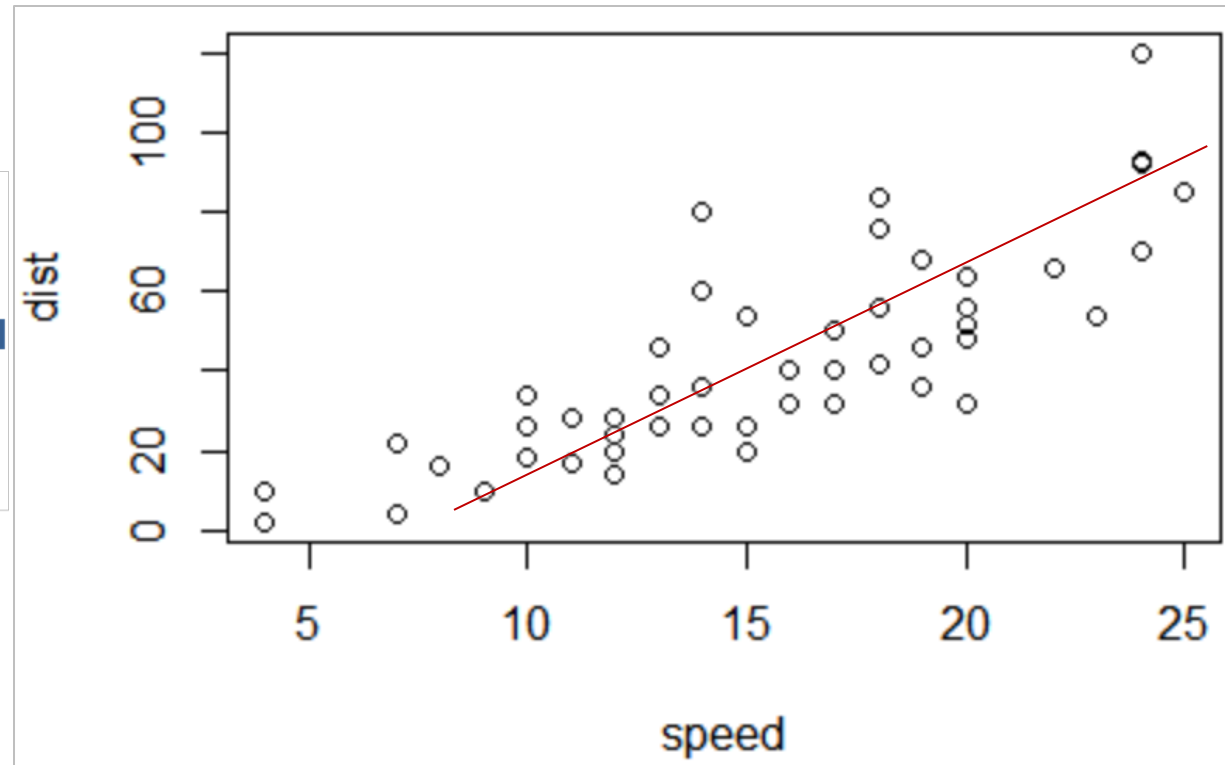
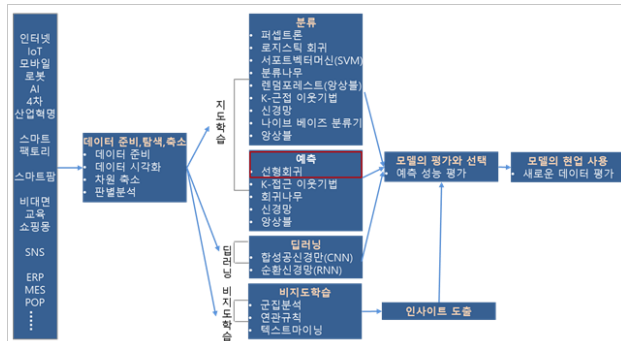
- 이제 실제 데이터를 가지고 모델링과 예측을 해보자
 - cars 는 자동차의 속도에 따른 제동 거리를 기록한 data
 - 베이스 R이 제공하는 cars 데이터 사용
 - 먼저 str(구조 확인)과 head(data 몇 건 보기) 함수로 데이터 내용을 확인

```
Console C:/RSources/
> str(cars)
'data.frame':  50 obs. of  2 variables:
 $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
 $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
> head(cars,10)
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
7    10   18
8    10   26
9    10   34
10   11   17
```

observation

7.4 단순 선형 회귀의 적용 : cars data

- 이제 실제 데이터를 가지고 모델링과 예측을 해보자
 - 모델링 전에 plot으로 가시화를 하자
 - > plot(cars)



7.4 단순 선형 회귀의 적용 : cars data

■ 모델 적합 (설명 변수와 반응 변수 정하기)

- speed(속도를 나타내는 변수)와 dist(제동 거리를 나타내는 변수) 중에 설명 변수와 반응 변수 정하기 ← R이 자동으로 해줄 수 없음.
- 데이터 과학자가 주어진 임무를 이해하고 결정해야 함
- 변수 사이의 원인과 결과 관계를 따짐 (**설명 변수와 반응 변수는 ?**)
- 원인에 해당하는 speed를 설명 변수로 함



cars 데이터에서 설명변수(독립변수) 반응 변수(종속 변수) 결정

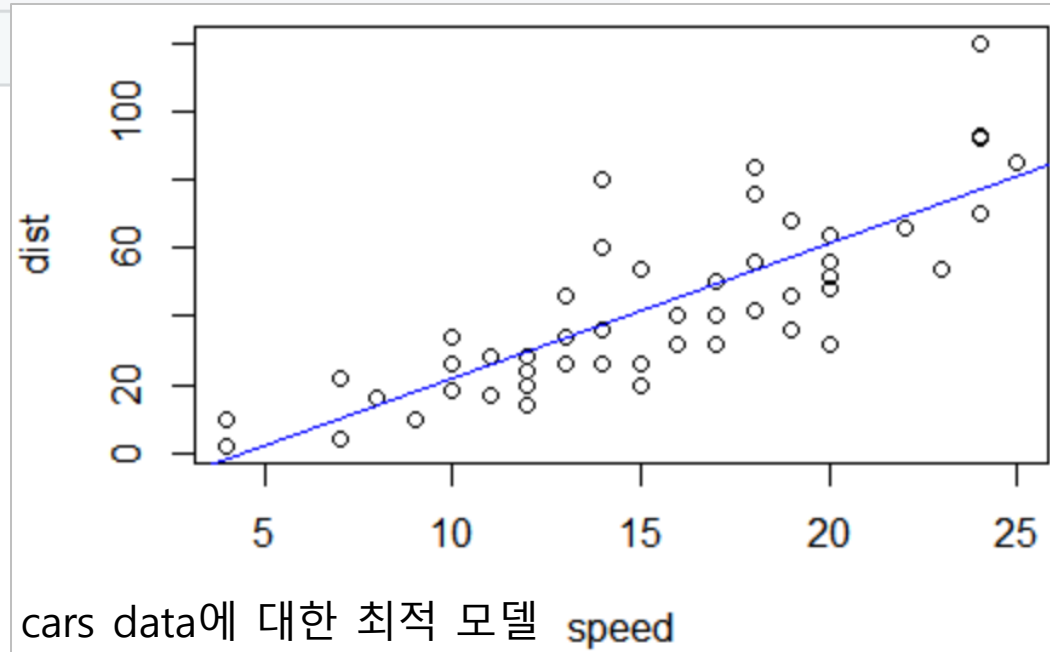
- 이처럼 설명 변수가 하나 뿐인 경우를 단순 선형 회귀라 함

7.4 단순 선형 회귀의 적용 : cars data

■ 모델 적합

Console C:/RSources/ ➡

```
> car_model = lm(dist~speed,data=cars)
> coef(car_model)
(Intercept)      speed
-17.579095      3.932409
> abline(car_model,col='blue')
```



최적 모델 : $\text{dist} = -17.579095 + 3.932409 \times \text{speed}$

7.4 단순 선형 회귀의 적용 : cars data

■ fitted 함수로 훈련 집합에 대한 예측 수행하기

- 네 번째 샘플을 관찰해 보면, speed:7, dist:22, 예측: 9.947766, 오차:12.052234
- 열 두 번째 data : speed :12, dist :14, 예측값 : 29.609810, 오차 : -15.609810

Console C:/RSource

```
> head(cars,15)
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
7    10   18
8    10   26
9    10   34
10   11   17
11   11   28
12   12   14
13   12   20
14   12   24
15   12   28
```

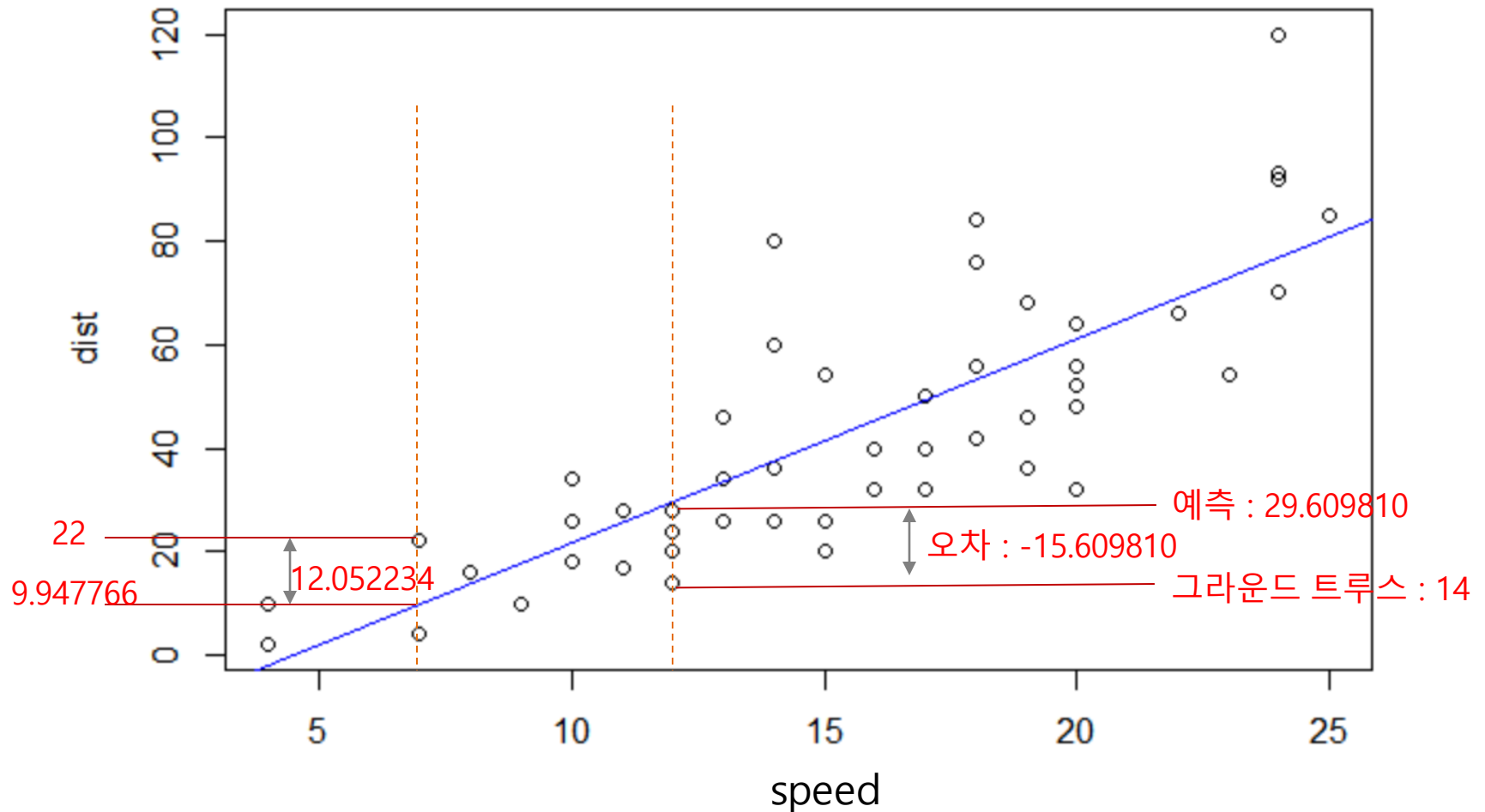
```
> fitted(car_model)
      1      2      3      4      5      6      7      8      9     10     11
-1.849460 -1.849460  9.947766  9.947766 13.880175 17.812584 21.744993 21.744993 21.744993 25.677401 25.677401
12      13      14      15      16      17      18      19      20      21      22
29.609810 29.609810 29.609810 29.609810 33.542219 33.542219 33.542219 33.542219 37.474628 37.474628 37.474628
23      24      25      26      27      28      29      30      31      32      33
37.474628 41.407036 41.407036 41.407036 45.339445 45.339445 49.271854 49.271854 49.271854 53.204263 53.204263
34      35      36      37      38      39      40      41      42      43      44
53.204263 53.204263 57.136672 57.136672 57.136672 61.069080 61.069080 61.069080 61.069080 61.069080 68.933898
45      46      47      48      49      50
72.866307 76.798715 76.798715 76.798715 76.798715 80.731124

> residuals(car_model)
      1      2      3      4      5      6      7      8      9     10
 3.849460 11.849460 -5.947766 12.052234  2.119825 -7.812584 -3.744993  4.255007 12.255007 -8.677401
11      12      13      14      15      16      17      18      19      20
 2.322599 -15.609810 -9.609810 -5.609810 -1.609810 -7.542219  0.457781  0.457781 12.457781 -11.474628
21      22      23      24      25      26      27      28      29      30
-1.474628 22.525372 42.525372 -21.407036 -15.407036 12.592964 -13.339445 -5.339445 -17.271854 -9.271854
31      32      33      34      35      36      37      38      39      40
 0.728146 -11.204263  2.795737 22.795737 30.795737 -21.136672 -11.136672 10.863328 -29.069080 -13.069080
41      42      43      44      45      46      47      48      49      50
-9.069080 -5.069080  2.930920 -2.933898 -18.866307 -6.798715 15.201285 16.201285 43.201285  4.268876
```

7.4 단순 선형 회귀의 적용 : cars data

■ fitted 함수로 훈련 집합에 대한 예측 수행하기

- 오차를 그래프에 표시하면



7.4 단순 선형 회귀의 적용 : cars data

■ 새로운 데이터에 대해 예측 해보기

- 예를 들어, 시속 21.5로 달리고 있었다면 제동 거리가 얼마일까?

```
> nx1 = data.frame(speed=c(21.5))  
> predict(car_model,nx1)
```

```
      1  
66.96769
```

- 시속 25부터 0.5씩 증가시키며 달렸을 때 제동 거리를 알고 싶다면,

```
> nx2=data.frame(speed=c(25.0,25.5,26.0,26.5,27.0,27.5,28.0))  
> predict(car_model,nx2)
```

```
      1      2      3      4      5      6      7  
80.73112 82.69733 84.66353 86.62974 88.59594 90.56215 92.52835
```

최적 모델 : $\text{dist} = -17.579095 + 3.932409 \times \text{speed}$

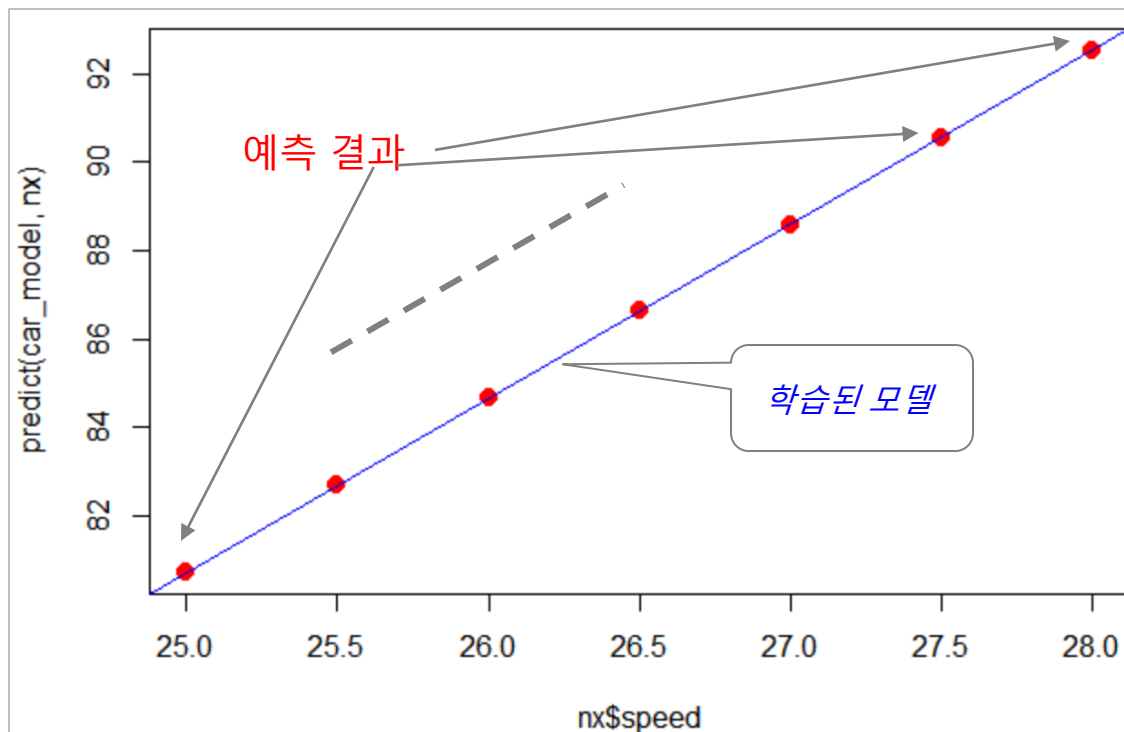
7.4 단순 선형 회귀의 적용 : cars data

■ 새로운 데이터에 대해 예측 해보기

- 그래프로 그려보면

Console C:/RSources/ ➞

```
> nx=data_frame(speed=c(25.0,25.5,26.0,26.5,27.0,27.5,28.0))  
> plot(nx$speed, predict(car_model, nx),col='red', cex=2, pch=20)  
> abline(car_model,col='blue')
```



7.4 단순 선형 회귀의 적용 : cars data

■ plot의 option

인 수	설 명
main = "메인 제목"	제목 설정
sub = "서브 제목"	서브 제목
xlab = "문자", ylab = "문자"	x, y축에 사용할 문자열을 지정합니다.
ann=F	x, y축 제목을 지정하지 않습니다.
tmag=2	제목 등에 사용되는 문자의 확대율 지정
axes=F	x, y축을 표시하지 않습니다.
axis	x, y축을 사용자의 지정값으로 표시합니다.
그래프 타입 선택	
type="p"	점 모양 그래프 (기본값)
type="l"	선 모양 그래프 (굵은선 그래프)
type="b"	점과 선 모양 그래프
type="c"	"b"에서 점을 생략한 모양
type="o"	점과 선을 중첩해서 그린 그래프
type="h"	각 점에서 x축 까지의 수직선 그래프
type="s"	왼쪽값을 기초로 계단모양으로 연결한 그래프
type="S"	오른쪽 값을 기초로 계단모양으로 연결한 그래프
type="n"	축만 그리고 그래프는 그리지 않습니다.
선의 모양 선택	
lty=0, lty="blank"	투명선
lty=1, lty="solid"	실선
lty=2, lty="dashed"	대쉬선
lty=3, lty="dotted"	점선
lty=4, lty="dotdash"	점선과 대쉬선
lty=5, lty="longdash"	긴 대쉬선
lty=6, lty="twodash"	2개의 대쉬선
색, 기호 등	
col=1, col="blue"	기호의 색지정, 1-검정, 2-빨강, 3-초록, 4-파랑, 5-연파랑, 6-보라, 7-노랑, 8-회색
pch=0, pch="문자"	점의 모양을 지정합니다
bg="blue"	그래프의 배경색 지정
lwd="숫자"	선을 그릴 때 선의 굵기를 지정
cex="숫자"	점이나 문자를 그릴 때 점이나 문자의 굵기를 지정

7.4 단순 선형 회귀의 적용 : cars data

■ lm에 고차 방정식 적용

- lm은 기본적으로 1차 방정식, 즉 직선으로 모델 적합
- poly 옵션을 사용하면 고차 방정식 적용 가능

Console C:/RSources/ 

```
> plot(cars,xlab='속도', ylab="거리")
>
> x=seq(0,25,length.out=200) # 예측할 지점
>
> for (i in 1:4) {
+   m=lm(dist~poly(speed, i), data=cars)
+   assign(paste('m', i, sep='.'),m) # i차 모델 m을 m.i라 함
+   lines(x, predict(m,data.frame(speed=x)),col=i) # m예측 결과 그림 그리기
+ }
```

```
> x
[1] 0.0000000 0.1256281 0.2512563 0.3768844 0.5025126 0.6281407
[7] 0.7537688 0.8793970 1.0050251 1.1306533 1.2562814 1.3819095
[13] 1.5075377 1.6331658 1.7587940 1.8844221 2.0100503 2.1356784
[19] 2.2613065 2.3869347 2.5125628 2.6381910 2.7638191 2.8894472
.....
[187] 23.3668342 23.4924623 23.6180905 23.7437186 23.8693467 23.9949749
[193] 24.1206030 24.2462312 24.3718593 24.4974874 24.6231156 24.7487437
[199] 24.8743719 25.0000000
```

7.4 단순 선형 회귀의 적용 : cars data

■ lm에 고차 방정식 적용

- lm은 기본적으로 1차 방정식, 즉 직선으로 모델 적합
- poly 옵션을 사용하면 고차 방정식 적용 가능
- for 문의 변형

```
m1=lm(dist~poly(speed, 1), data=cars)
lines(x, predict(m1,data.frame(speed=x)),col=1)
```

```
m2=lm(dist~poly(speed, 2), data=cars)
lines(x, predict(m2,data.frame(speed=x)),col=2)
```

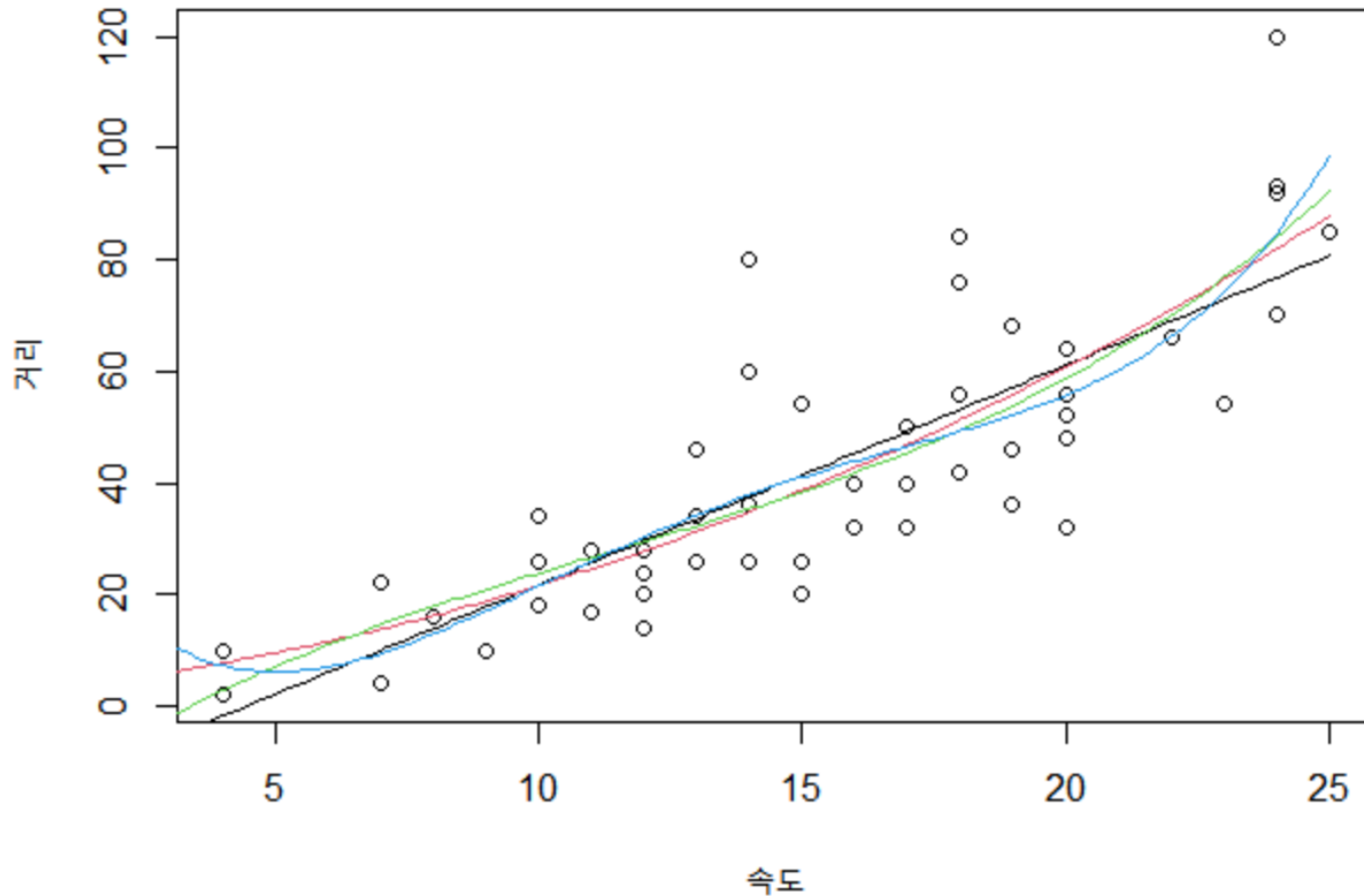
```
m3=lm(dist~poly(speed, 3), data=cars)
lines(x, predict(m3,data.frame(speed=x)),col=3)
```

```
m4=lm(dist~poly(speed, 4), data=cars)
lines(x, predict(m4,data.frame(speed=x)),col=4)
```

```
anova(m1, m2, m3, m4)
```

7.4 단순 선형 회귀의 적용 : cars data

- 고차 다항식 적용과 분산 분석(ANOVA)
- lm에 고차 방정식 적용

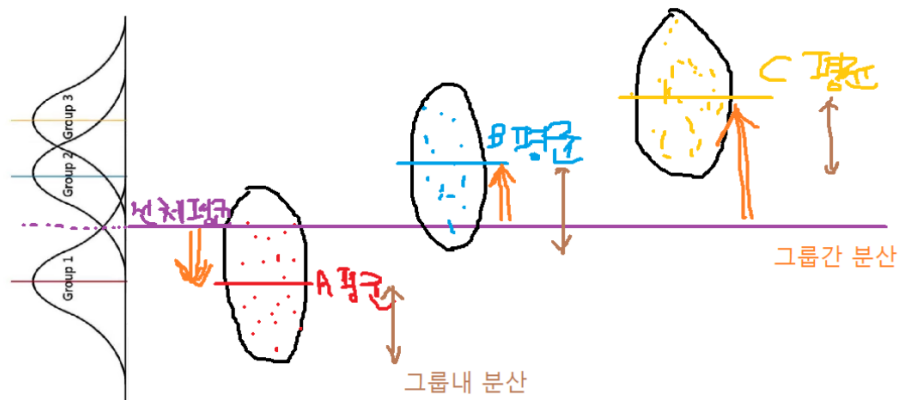


1,2,3,4차 방정식을 이용한 모델

7.4 단순 선형 회귀의 적용 : cars data

■ 분산 분석(ANOVA(ANalysis Of VAriance))?

분산 분석(analysis of variance, ANOVA, 변량 분석)은 통계학에서 두 개 이상 다수의 집단을 비교하고자 할 때 집단 내의 분산, 총평균과 각 집단의 평균의 차이에 의해 생긴 집단 간 분산의 비교를 통해 만들어진 F분포를 이용하여 가설검정을 하는 방법이다.



7.4 단순 선형 회귀의 적용 : cars data

- anova 함수로 분산 분석(ANOVA, ANalysis Of VAriance) 해보기
 - 여러 모델 간에 차이가 있는지에 대해 통계적 유의성을 확인해 줌
 - 이 예제에서는 $\Pr(>F)$, 즉 p-값은 모두 0.05보다 커서 통계적으로 차이가 없다고 판정할 수 있음 → 가장 단순한 1차 모델, 즉 m.1을 사용하는 것이 현명

Console C:/RSources/ ↗

```
> # 분산 분석
> anova(m.1, m.2, m.3, m.4)
Analysis of Variance Table

Model 1: dist ~ poly(speed, i)
Model 2: dist ~ poly(speed, i)
Model 3: dist ~ poly(speed, i)
Model 4: dist ~ poly(speed, i)
  Res.Df  RSS Df Sum of Sq    F Pr(>F)
1      48 11354      528.81 2.3108 0.1355
2      47 10825      190.35 0.8318 0.3666
3      46 10634      336.55 1.4707 0.2316
4      45 10298
```


Thank you

