1번

다음은 사내 헬스 클럽의 정규 회원 30명에 대한 건강 기록에 관한 것이다. x1 = 몸무게 (단위: 파운드) x2 = 휴식시 1분당 맥박수 x3 = 팔, 다리의 힘(strength)(들어올릴 수 있는 파운드 수) x4 = 1/4-mile trial run 시 걸린 시간(단위: 초) y = 1-mile run 시 걸린 시간 (단위: 초)

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} + u_i, \quad u_i \sim N(0, \sigma^2), \quad i = 1, \cdots, 30$$

을 적합하고 여러 가지 진단 통계량에 의해서 outlier, influential points, high leverage points가 있는지 알아보아라.(이들 통계량들의 값에 대한 표와 plot을 제시할 것!)

In [ ]:
```
x1 = c(217, 141, 152, 153, 180, 193, 162, 180, 205, 168,
       232, 146, 173, 155, 212, 138, 147, 197, 165, 125,
       161, 132, 257, 236, 149, 161, 198, 245, 141, 177)
x2 = c(67, 52, 58, 56, 66, 71, 65, 80, 77, 74,
       65, 68, 51, 64, 66, 70, 54, 76, 59, 58,
       52, 62, 64, 72, 57, 57, 59, 70, 63, 53)
x3 = c(260, 190, 203, 183, 170, 178, 160, 170, 188, 170,
       220, 158, 243, 198, 220, 180, 150, 228, 188, 160,
       190, 163, 313, 225, 173, 173, 220, 218, 193, 183)
x4 = c(91, 66, 68, 70, 77, 82, 74, 84, 83, 79,
       72, 68, 56, 59, 77, 62, 75, 88, 70, 66,
       69, 59, 96, 84, 68, 65, 62, 69, 60, 75)
y = c(481, 292, 338, 357, 396, 429, 345, 469, 425, 358,
      393, 346, 279, 311, 401, 267, 404, 442, 368, 295,
      391, 264, 487, 481, 374, 309, 367, 469, 252, 338)
```

주어진 데이터를
$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} + u_i, \quad u_i \sim N(0, \sigma^2), \quad i = 1, \cdots, 30$$
로 적합하는 R 코드는 다음과 같다.

In [ ]:
```
df = data.frame(x1, x2, x3, x4, y)
result = lm(y~x1+x2+x3+x4, data=df)
summary(result)
```

```
Call:
lm(formula = y ~ x1 + x2 + x3 + x4, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-55.223 -18.821  -5.321  18.928  44.487

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.6186    56.1027  -0.064 0.949086
x1            1.2676     0.2869   4.419 0.000168 ***
x2           -0.5252     0.8628  -0.609 0.548194
x3           -0.5050     0.2459  -2.054 0.050614 .
x4            3.9030     0.7477   5.220 2.11e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 28.67 on 25 degrees of freedom
Multiple R-squared:  0.8531,    Adjusted R-squared:  0.8296
F-statistic:  36.3 on 4 and 25 DF,  p-value: 4.51e-10
```

적합된 식은 다음과 같다.

$$y_i = 1.268 \cdot x_1 + (-0.525) \cdot x_2 + (-0.505) \cdot x_3 + 3.903 \cdot x_4 - 3.619$$

outlier 인지 여부는 Studentized Residual로 판단할 수 있는데, Studentized Residual을 구하기 위해서는 우선 Hat Value를 구해야 한다.

참고로 Hat Value가 Hat value의 평균보다 크면 Leverage가 크다고 할 수 있다.

Hat Value의 평균은 $\bar{h} = \frac{k+1}{n} = \frac{4+1}{30} = 0.167$ 이다.

따라서 Hat Value가 0.167보다 큰 데이터는 Leverage가 크다고 할 수 있다.

In [ ]:
```r
# Hat_value의 평균 구하기(Hat_value의 평균은 (k+1)/n 임을 확인)
mean_hat = mean(hatvalues(result))

# df에 HatValue 컬럼 추가
df$HatValue = hatvalues(result)

# df에 IsLeverageHigh 컬럼 추가
leverage = ifelse(hatvalues(result) > mean_hat, "Y", "N")
df$IsLeverageHigh = leverage

# IsLeverageHigh 값의 인덱스 확인
which(df$IsLeverageHigh == "Y")
```

1 · 8 · 11 · 13 · 16 · 17 · 18 · 23 · 27 · 28 · 30

1, 8, 11, 13, 16, 17, 18, 23, 27, 28, 30 번째 데이터는 각각의 Hat Value가 Hat Value의 평균인 0.167보다 크기 때문에 Leverage가 큰 데이터라고 할 수 있다.

Outlier인지 여부는 Studentized Residual 로 판단할 수 있다.

자기자신을 제외한 데이터로 계산한 External Studentized Residual을 사용하며, 자기자신을 제외한 데이터로 여러 번의 계산을 하는 것을 다중 비교로 보아 본페로니 적응 방

법을 사용하여 검정할 필요가 있다.

In [ ]:
```
# Studentized Residual 구하기

# 자기자신을 포함하지 않은 External Studentized Residual은 rstudent 함수로 구한다
df$StuRes = rstudent(result)

# 자기자신을 포함한 Internal Studentized Residual은 rstandard 함수로 구한다.
# (비교를 위해 계산해 본다)
df$StuResForCompare = rstandard(result)
```

In [ ]:
```
# t_max를 구하기 위해 df$StuRes에서 가장 절대값이 큰 값을 찾고, 이 값으로 p-value
t_max = max(abs(df$StuRes))
cat("가장 큰 t 절대값:", t_max, "\n")

# t분포 검정
p_single = pt(t_max, df=30-4-2, lower.tail = FALSE)
cat("단측검정 p값: ", p_single, "\n")
p_two_tailed = 2*p_single
cat("양측검정 p값: ", p_two_tailed, "\n")

# 30번의 관찰을 한 경우 본페로니 적응을 적용하면 p 값은 다음과 같다.
p_final = p_two_tailed / 30
cat("본페로니 적응 p값:", p_final)
```

```
가장 큰 t 절대값: 2.325376
단측검정 p값:  0.01441446
양측검정 p값:  0.02882892
본페로니 적응 p값: 0.000960964양측검정 p값:  0.02882892
본페로니 적응 p값: 0.000960964
```

In [ ]:
```
# outlier가 되는 t값은 절대값이 3.483보다 큰 경우라고 할 수 있다.
qt(0.000960964, 24, lower.tail = FALSE)

# outlier 여부를 판단해서 IsOutlier 컬럼으로 추가한다.
outlier = ifelse(abs(df$StuRes) > 3.483, "Y", "N")
df$IsOutlier = outlier
```

3.48289916130436

In [ ]:
```
# 유의수준 0.05 수준으로 Outlier가 있는지도 살펴본다.
qt(0.025, 24, lower.tail=FALSE)

# outlier 여부를 판단해서 IsOutlier 컬럼으로 추가한다.
outlier_95 = ifelse(abs(df$StuRes) > 2.064, "Y", "N")
df$IsOutlier_95 = outlier_95
```

2.06389856162803

In [ ]:
```
df
```

A data.frame: 30 × 11

x1	x2	x3	x4	y	HatValue	IsLeverageHigh	StuRes	StuResFo
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>	<dbl>	
217	67	260	91	481	0.24940559	Y	0.8347688	
141	52	190	66	292	0.12621605	N	-0.6434769	-
152	58	203	68	338	0.09037161	N	0.5961034	
153	56	183	70	357	0.07867432	N	0.5477671	
180	66	170	77	396	0.09034233	N	-0.3076680	-
193	71	178	82	429	0.11042797	N	-0.1775854	-
162	65	160	74	345	0.08431676	N	-1.1217615	-
180	80	170	84	469	0.22058648	Y	1.8381930	
205	77	188	83	425	0.14679644	N	-0.7412875	-
168	74	170	79	358	0.12857122	N	-1.3259496	-
232	65	220	72	393	0.21353297	Y	-1.3272900	-
146	68	158	68	346	0.09929043	N	0.5304770	
173	51	243	56	279	0.29756398	Y	-0.2345047	-
155	64	198	59	311	0.16159997	N	0.8120482	
212	66	220	77	401	0.07042503	N	-0.6756303	-
138	70	180	62	267	0.23146500	Y	-0.7344577	-
147	54	150	75	404	0.24815163	Y	1.3342612	
197	76	228	88	442	0.22665324	Y	0.2915908	
165	59	188	70	368	0.04660222	N	0.5344113	
125	58	160	66	295	0.11399262	N	-0.2241964	-
161	52	190	69	391	0.11933222	N	1.7165448	
132	62	163	59	264	0.12171025	N	-0.5543192	-
257	64	313	96	487	0.51331861	Y	-0.9049781	-
236	72	225	84	481	0.14319481	N	0.3348513	
149	57	173	68	374	0.07214601	N	1.5086721	
161	57	173	65	309	0.08447846	N	-1.0162254	-
198	59	220	62	367	0.16696906	Y	0.7473628	
245	70	218	69	469	0.38750148	Y	1.8480643	
141	63	193	60	252	0.16336900	N	-1.0205245	-
177	53	183	75	338	0.19299424	Y	-2.3253761	-

External Studentized Residual이 본페로니 적응 방법을 적용하여 계산한 t_max의 p_value 값을 충족시키는 값이 없으므로, 결론적으로 Outlier는 없다.

다만, alpha = 0.05 수준에서는 30번째 데이터를 Outlier로 볼 수 있다.

Influential Points 여부는 쿡의 거리를 통해 판단할 수 있다.

$D_i > \frac{4}{n-k-1}$ 이면 Influential Points로 판단할 수 있다.

In [ ]:
```
# 쿡의 거리를 계산하여 CookDis 컬럼에 입력한다.
df$CookDis = cooks.distance(result)

# 쿡의 거리가 4/(30-4-1) = 0.16 보다 크면 IsInfluential 에 "Y"로 입력한다.
df$IsInfluential = ifelse(df$CookDis > 0.16, "Y", "N")

# IsLeverageHigh 값의 인덱스 확인
which(df$IsInfluential == "Y")
```

8 · 23 · 28 · 30

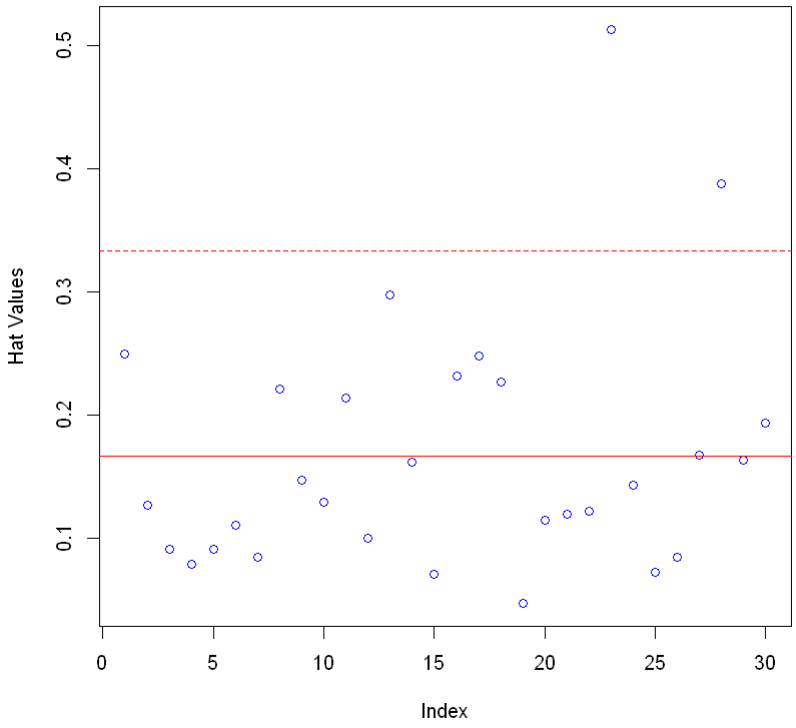
쿡의 거리가 0.16 이상이어서 영향력이 있는 데이터는 8, 23, 28, 30 번째 데이터 이다.

In [ ]:
```
df
```

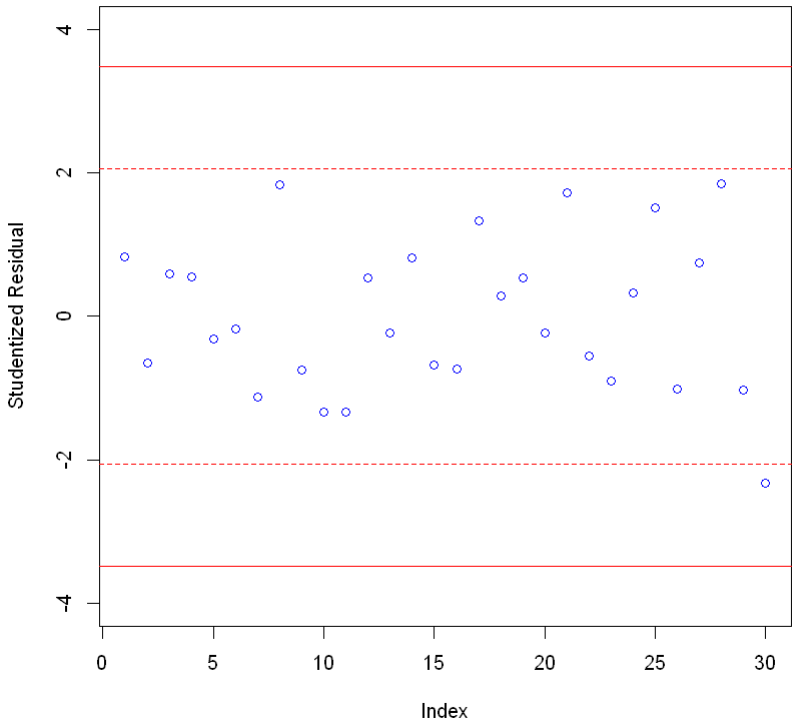
A data.frame: 30 × 13

x1	x2	x3	x4	y	HatValue	IsLeverageHigh	StuRes	StuResFo
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>	<dbl>	
217	67	260	91	481	0.24940559	Y	0.8347688	
141	52	190	66	292	0.12621605	N	-0.6434769	-
152	58	203	68	338	0.09037161	N	0.5961034	
153	56	183	70	357	0.07867432	N	0.5477671	
180	66	170	77	396	0.09034233	N	-0.3076680	-
193	71	178	82	429	0.11042797	N	-0.1775854	-
162	65	160	74	345	0.08431676	N	-1.1217615	-
180	80	170	84	469	0.22058648	Y	1.8381930	
205	77	188	83	425	0.14679644	N	-0.7412875	-
168	74	170	79	358	0.12857122	N	-1.3259496	-
232	65	220	72	393	0.21353297	Y	-1.3272900	-
146	68	158	68	346	0.09929043	N	0.5304770	
173	51	243	56	279	0.29756398	Y	-0.2345047	-
155	64	198	59	311	0.16159997	N	0.8120482	
212	66	220	77	401	0.07042503	N	-0.6756303	-
138	70	180	62	267	0.23146500	Y	-0.7344577	-
147	54	150	75	404	0.24815163	Y	1.3342612	
197	76	228	88	442	0.22665324	Y	0.2915908	
165	59	188	70	368	0.04660222	N	0.5344113	
125	58	160	66	295	0.11399262	N	-0.2241964	-
161	52	190	69	391	0.11933222	N	1.7165448	
132	62	163	59	264	0.12171025	N	-0.5543192	-
257	64	313	96	487	0.51331861	Y	-0.9049781	-
236	72	225	84	481	0.14319481	N	0.3348513	
149	57	173	68	374	0.07214601	N	1.5086721	
161	57	173	65	309	0.08447846	N	-1.0162254	-
198	59	220	62	367	0.16696906	Y	0.7473628	
245	70	218	69	469	0.38750148	Y	1.8480643	
141	63	193	60	252	0.16336900	N	-1.0205245	-
177	53	183	75	338	0.19299424	Y	-2.3253761	-

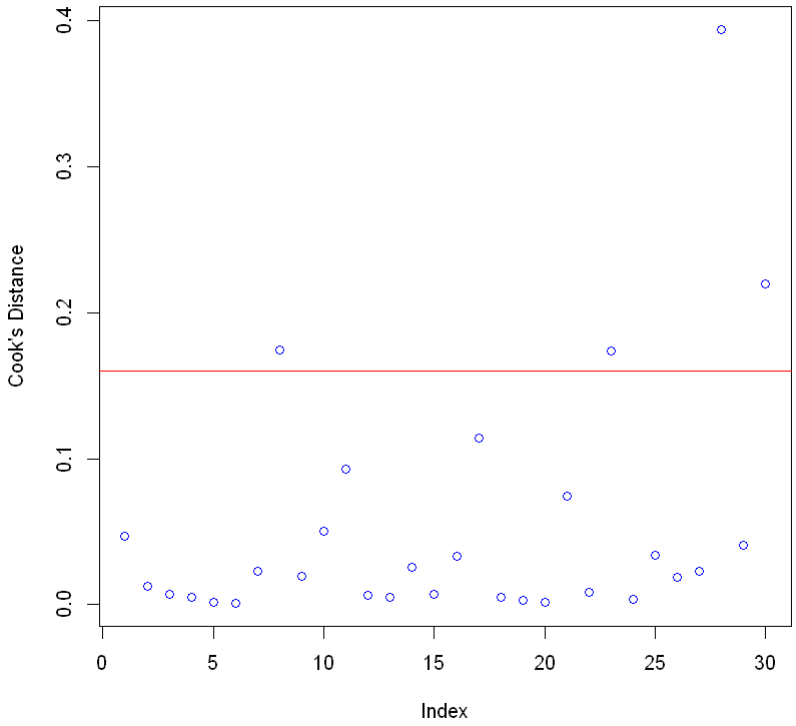
In [ ]:
```
# Hat Value 그래프
plot(df$HatValue, xlab="Index", ylab="Hat Values", col="blue")
abline(h=mean(df$HatValue), col="red")
abline(h=2*mean(df$HatValue), col="red", lty=2)
```



In [ ]:
```
# 잔차 그래프
plot(df$StuRes, ylim=c(-4,4), xlab="Index", ylab="Studentized Residual", col="bl
abline(h=3.483, col="red")
abline(h=-3.483, col="red")
abline(h=2.063, col="red", lty=2)
abline(h=-2.063, col="red", lty=2)
```



In [ ]:
```
# 영향력 그래프
plot(df$CookDis, xlab="Index", ylab="Cook's Distance", col="blue")
abline(h=0.16, col="red")
```



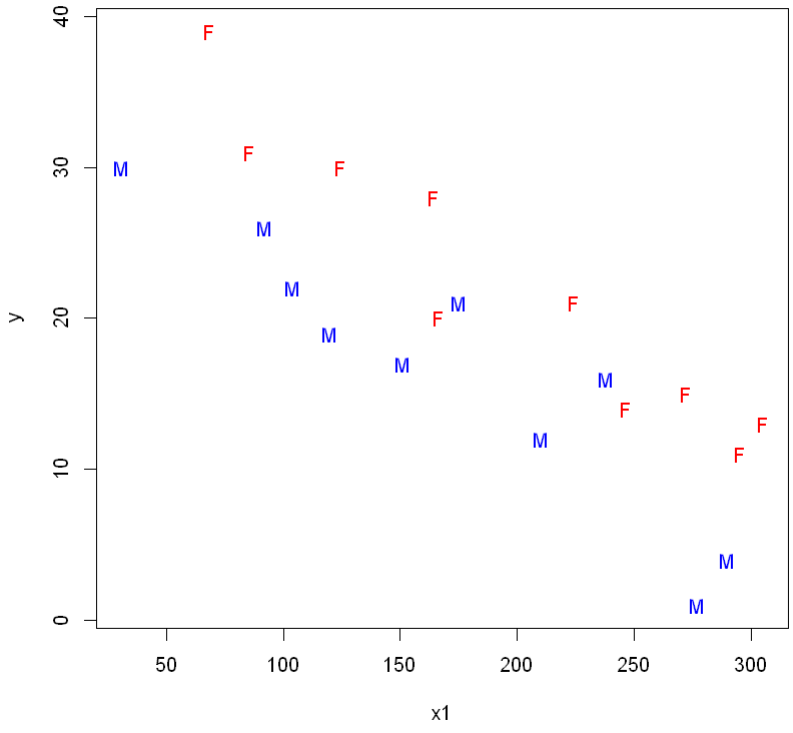
2번

어떤 전자회사에서 새로운 공정기법을 개발하여 작업자들에게 이 기법을 교육시키며 다음과 같은 자료를 얻었다.

In [ ]:
```
ID = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
       11, 12, 13, 14, 15, 16, 17, 18, 19, 20)

# 시간(y)
time = c(17, 26, 21, 30, 22, 1, 12, 19, 4, 16,
         28, 15, 11, 39, 31, 21, 20, 13, 30, 14)

# 점수(x1)
score = c(151, 92, 175, 31, 104, 277, 210, 120, 290, 238,
          164, 272, 295, 68, 85, 224, 166, 305, 124, 246)

# 성별(x2)
sex = c("남", "남", "남", "남", "남", "남", "남", "남", "남", "남",
        "여", "여", "여", "여", "여", "여", "여", "여", "여", "여")

edu = c("중졸", "초졸", "중졸", "초졸", "초졸", "고졸", "고졸", "중졸", "고졸", "
        "고졸", "고졸", "고졸", "초졸", "중졸", "고졸", "중졸", "고졸", "중졸", "

# 시간: 새로운 공정기법을 습득할 때까지 소요된 교육시간
# 점수: 교육시키기 전에 측정한 적성검사점수
```

(1) y와 x1의 산점도를 그리되 남자의 좌표는 'M', 여자의 좌표는 'F'로 나타내어라.

In [ ]:
```
plot(score, time, type = "n", xlab = "x1", ylab = "y")  # 빈 플롯 생성
points(score[sex == "남"], time[sex == "남"], pch = "M", col = "blue")  # 남자 데
points(score[sex == "여"], time[sex == "여"], pch = "F", col = "red")  # 여자 데
```



(2) 각 성별로 회귀모형

$y_i = \beta_0 + \beta_1 x_{1i} + u_i$ (남자)

$y_i = \beta_0' + \beta_1' x_{1i} + u_i$ (여자)
을 적합시켜라.

In [ ]:
```
# 남자 데이터 추출
male_data = subset(data.frame(ID, time, score, sex), sex=="남")
# 모형 적합
male_model = lm(time ~ score, data=male_data)
summary(male_model)

# 여자 데이터 추출
female_data = subset(data.frame(ID, time, score, sex), sex=="여")
# 모형 적합
female_model = lm(time ~ score, data=female_data)
summary(female_model)
```

```
Call:
lm(formula = time ~ score, data = male_data)

Residuals:
    Min      1Q  Median      3Q     Max
-4.9953 -1.5008 -0.6915  1.0080  6.1102

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 33.65610    2.60379  12.926 1.21e-06 ***
score       -0.09986    0.01393  -7.171 9.51e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.542 on 8 degrees of freedom
Multiple R-squared:  0.8654,    Adjusted R-squared:  0.8485
F-statistic: 51.43 on 1 and 8 DF,  p-value: 9.51e-05
Call:
lm(formula = time ~ score, data = female_data)

Residuals:
    Min      1Q  Median      3Q     Max
-5.2014 -2.1613  0.6219  2.1314  3.6208

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 42.44130    2.45512  17.287 1.28e-07 ***
score       -0.10385    0.01162  -8.938 1.95e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.998 on 8 degrees of freedom
Multiple R-squared:  0.909,     Adjusted R-squared:  0.8976
F-statistic: 79.88 on 1 and 8 DF,  p-value: 1.95e-05
```

남성의 적합된 회귀모형은 $\hat{y}_i = 33.656 + (-0.100)\hat{x_{1i}}$

여성의 적합된 회귀모형은 $\hat{y}_i = 42.441 + (-0.104)\hat{x_{1i}}$

(3) 가변수(dummy variable) x2를 x2 = 0 (남), x2 = 1 (여) 와 같이 정의할 때 회귀모형 $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + u_i$를 적합시키고 (2)의 결과와 비교하여라.

In [ ]:
```
# 가변수 x2 생성
x2 = ifelse(sex=="남", 0, 1)

# 데이터 프레임 생성
data = data.frame(time, score, x2)

# 회귀모형 적합
model = lm(time ~ score + x2, data=data)
summary(model)
```

```
Call:
lm(formula = time ~ score + x2, data = data)

Residuals:
   Min     1Q Median     3Q    Max
-5.144 -1.809 -0.527  1.828  6.250

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  33.9981     1.7952  18.938 7.28e-13 ***
score        -0.1019     0.0088 -11.578 1.74e-09 ***
x2            8.0592     1.4441   5.581 3.31e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.188 on 17 degrees of freedom
Multiple R-squared:  0.8972,    Adjusted R-squared:  0.8851
F-statistic:  74.2 on 2 and 17 DF,  p-value: 3.993e-09
```

적합된 회귀모형은 $\hat{y}_i = 33.998 + (-0.102)\hat{x_{1i}} + 8.059\hat{x_{2i}}$ 이다.

x2 = 0일 때는 $\hat{y}_i = 33.998 + (-0.102)\hat{x_{1i}}$

x2 = 1일 때는 $\hat{y}_i = 42.057 + (-0.102)\hat{x_{1i}}$ 이 된다.

이 결과는 남녀 각각의 데이터를 적합한 식과 상당히 유사하다.

(5) 위 (3)의 모형 $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + u_i$에 학력을 독립변수로 추가하려면 0 또는 1의 값을 가지는 가변수를 몇 개 만들어야 하는가? 필요한 개수만큼 가변수를 넣어 회귀모형을 적합시켜라.

학력은 "초졸", "중졸", "고졸"의 세 가지 범주를 가지기 때문에 2개의 가변수를 사용해야 한다. "초졸"은 기본값으로 보고, "중졸"을 나타내는 변수, "고졸"을 나타내는 변수를 생성하면 된다.

edu1: "중졸"일 때 1, 나머지는 0 edu2: "고졸"일 때 1, 나머지는 0

(edu1, edu2)가 (0,0)이면 "초졸", (1,0)이면 "중졸", (0,1)이면 "고졸"을 나타낸다.

In [ ]:
```
# 학력에 대한 가변수 생성
edu1 = ifelse(edu == "중졸", 1, 0)
edu2 = ifelse(edu == "고졸", 1, 0)

# 데이터 프레임 생성
data = data.frame(time, score, x2, edu1, edu2)

# 회귀모형 적합
```

```
model = lm(time ~ score + x2 + edu1 + edu2, data=data)
summary(model)
```

```
Call:
lm(formula = time ~ score + x2 + edu1 + edu2, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.9231 -1.8429 -0.4515  1.5630  6.2307 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 35.35072    2.28529  15.469 1.26e-10 ***
score       -0.11005    0.02032  -5.416 7.15e-05 ***
x2           8.06112    1.53101   5.265 9.51e-05 ***
edu1        -1.32312    2.52588  -0.524    0.608    
edu2         1.05556    4.20624   0.251    0.805    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.271 on 15 degrees of freedom
Multiple R-squared:  0.9046,    Adjusted R-squared:  0.8791 
F-statistic: 35.54 on 4 and 15 DF,  p-value: 1.735e-07
```

적합된 식은 다음과 같다.

$$\hat{y_i} = 35.351 + (-0.110)\hat{x_{1i}} + 8.06\hat{x_{2i}} + (-1.323)\hat{x_{3i}} + 1.056\hat{x_{4i}}$$

x1: 시간

x2: 성별

x3: 학력 가변수 edu1 - 중졸

x4: 학력 가변수 edu2 - 고졸