11.1 최소제곱법(The Method of Least Squares)

§. 직선접합(Fitting a Straight Line)

Theorem 11.1.1 최소제곱(Least Squares)
$(x_1,y_1),\ldots,(x_n,y_n)$을 n개의 점들의 집합이라 하자. 모든 점 $(x_1,y_1),\ldots,(x_n,y_n)$에 대하여 가장 가까운 직선, 즉 각 점과 직선 사이의 거리를 최소화하는 직선은 다음과 같은 기울기와 절편을 갖는다.

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(y_i-\bar{y})(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2} \ , \ \hat{\beta}_0 = \bar{y}-\hat{\beta}_1\bar{x} \ \text{ where } \ \bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i, \ \bar{y}=\frac{1}{n}\sum_{i=1}^{n}y_i \ .$$

(Proof) 임의의 직선 $y=\beta_0+\beta_1 x$를 고려하자. 이 때 변수 x를 제외한 나머지는 고정된 상수이다. $x=x_i$로 두어 각 y_i에 대한 직선 접합을 고려할 것이다. 이제 다음과 같은 직선 y와 점 (x_i,y_i) 사이의 수직 거리에 관한 n개 제곱합을 생각하자.

$$Q=\sum_{i=1}^{n}\left[(y_i-(\beta_0+\beta_1 x_i)\right]^2$$

이러한 제곱합 Q를 β_0과 β_1에 관하여 최소화하고자 한다. 이는 Q를 β_0과 β_1에 관하여 편미분한 식을 0으로 두고 방정식을 푸는 것과 같다.

$$\frac{\partial Q}{\partial \beta_0}=-2\sum_{i=1}^{n}(y_i-\beta_0-\beta_1 x_i)=0 \ , \ \frac{\partial Q}{\partial \beta_1}=-2\sum_{i=1}^{n}(y_i-\beta_0-\beta_1 x_i)x_i=0$$

위의 편미분된 방정식을 풀면, 다음과 같은 방정식계를 얻는다.

$$\beta_0 n+\beta_1\sum_{i=1}^{n}x_i=\sum_{i=1}^{n}y_i$$
$$\beta_0\sum_{i=1}^{n}x_i+\beta_1\sum_{i=1}^{n}x_i^2=\sum_{i=1}^{n}x_i y_i$$

위와 같은 방정식계를 β_0과 β_1에 관한 ***정규방정식(normal equation)***이라 한다. 이제 Q에 대한 편미분을 한 번 더 진행하여 위 정규방정식을 만족하는 β_0과 β_1을 구하면 그러한 β_0과 β_1가 Q를 최소화하는 값임을 찾을 수 있고, 따라서 정리 11.1.1의 결과를 얻는다.

▲

Definition 11.1.1 최소제곱선(Least-Squares Line)
최소제곱법을 적용하여 얻은 β_0과 β_1를 각각 $\hat{\beta}_0$, $\hat{\beta}_1$라 두고, 이러한 $\hat{\beta}_0$, $\hat{\beta}_1$에 관한 직선의 방정식 $y=\hat{\beta}_0+\hat{\beta}_1 x$를 **최소제곱선**이라 부른다.

▷ **예제 11.1.1~2 : Blood Pressure.**

10명의 환자가 두 개의 다른 혈압에 관한 약품을 동일한 양으로 투여 받았다고 가정하자. 첫 번째로 일반 의약품인 A를 투여한 후 혈압을 측정하였고, 약의 효능이 다한 후 신약인 B를 투여한 후 혈압을 측정하여 이를 기록하였다. 이 때 각 환자가 약품을 투여 받고 변화된 혈압의 변화를 반응(reaction)이라 한다. 우리는 이러한 반응을 예측하고 싶어 한다. 자세하게 말하면, 기존에 알려진 약물 A의 반응을 가지고 약물 B의 반응을 예측하고 싶다. 두 약물에 대한 반응을 기록한 표는 아래와 같이 주어진다.

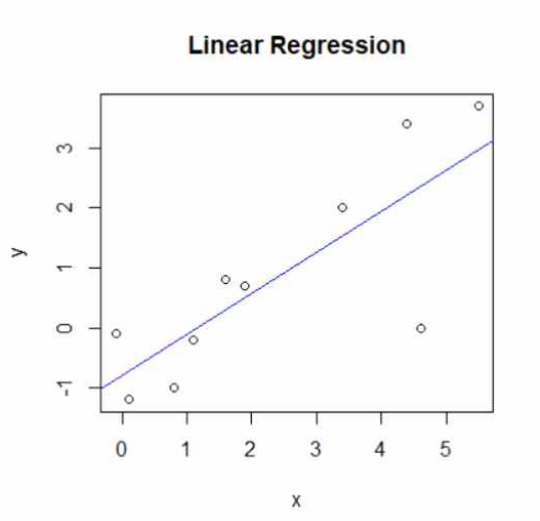
Table 11.1 Reactions to two drugs

i	x_i	y_i
1	1.9	0.7
2	0.8	-1.0
3	1.1	-0.2
4	0.1	-1.2
5	-0.1	-0.1
6	4.4	3.4
7	4.6	0.0
8	1.6	0.8
9	5.5	3.7
10	3.4	2.0

Table 11.1로부터, 전체 관측 수 $n=10$이고, 정리 11.1.1에 의해 $\hat{\beta}_0=-0.786$, $\hat{\beta}_1=0.685$으로 구할 수 있다. 따라서 이 관측값들에 대한 최소제곱선은 $y=-0.786+0.685x$로 구해진다.

R Code in Example 11.1.1~2

Results




```
> model
Coefficients:
(Intercept)            x
    -0.7861       0.6850
```

R Code

```
# Table
drug_A<-c(1.9,0.8,1.1,0.1,-0.1,4.4,4.6,1.6,5.5,3.4)
drug_B<-c(0.7,-1.0,-0.2,-1.2,-0.1,3.4,0.0,0.8,3.7,2.0)
df_blood=data.frame(drug_A,drug_B)
df_blood

# Variables
x=df_blood$drug_A
y=df_blood$drug_B

# Simple Linear Model
model=lm(y ~ x)
plot(x,y,main="Linear Regression")
abline(model,col="blue")
```

§. 최소제곱법을 이용한 다항식 접합 (Fitting a Polynomial by the Method of Least Squares)

앞서 논의하였던 직선접합에 대한 일반화로, 최소제곱법을 이용하여 다항식의 경우도 일반화할 수 있다. 즉, $y=\beta_0+\beta_1 x+\beta_2 x^2+\cdots+\beta_k x^k$로 일반화할 수 있다. 이를 구하는 방법은, 마찬가지로 점과 임의의 다항식 사이의 제곱합 Q를 최소화하는 계수 값을 찾으면 된다. 이 때 정규방정식은 $k+1$개의 Q에 대한 편미분을 푸는 것으로써, $k+1 \times k+1$ 행렬이 된다. 일련의 과정을 거치면, 최소제곱법을 이용한 다항식 접합을 $y=\hat{\beta}_0+\hat{\beta}_1 x+\hat{\beta}_2 x^2+\cdots+\hat{\beta}_k x^k$ 로 얻을 수 있다.

▷ **예제 11.1.3 : Fitting a Parabola.**

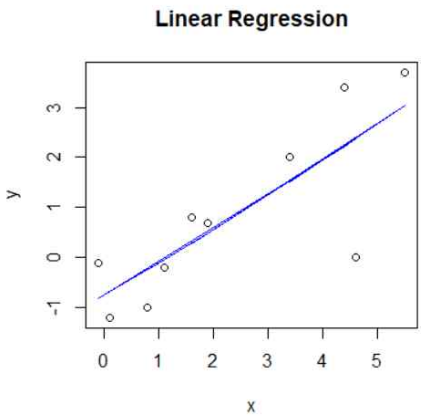
앞선 혈압과 약물 복용에 대한 예제 11.1.1을 상기하자. 이전 예제에서 우리는 산점도에 표시된 점들을 최적근사하는 선을 직선접합으로 택하였다. 이번 예제에서는 이를 곡선접합, 즉 다항식 접합으로 근사해보고자 한다. 다항식은 포물선의 형태, $y=\beta_0+\beta_1 x+\beta_2 x^2$로 근사하고자 한다. 이전 예제에서 했던 가정들을 그대로 이용하여, 최소제곱해인 각 β_i에 대한 값들을 구하려면 다음 정규방정식을 풀어야한다.

$$
\begin{aligned}
10\beta_0+23.3\beta_1+90.37\beta_2 &= 8.1 \\
23.3\beta_0+90.37\beta_1+401.0\beta_2 &= 43.59 \\
90.37\beta_0+401.0\beta_1+1892.7\beta_2 &= 204.55
\end{aligned}
$$

이러한 정규방정식을 만족하는 각 β_i의 값은 $\hat{\beta_0}=-0.744$, $\hat{\beta_1}=0.616$, $\hat{\beta}_2=0.013$으로 구할 수 있다. 따라서 최소제곱 포물선 $y=-0.744+0.616x+0.013x^2$ 을 얻는다.

R Code in Example 11.1.3

Results




```
> model

Coefficients:
(Intercept)          x      new_x
    -0.7451     0.6186     0.0126
```

R Code

```
# Table
drug_A<-c(1.9,0.8,1.1,0.1,-0.1,4.4,4.6,1.6,5.5,3.4)
drug_B<-c(0.7,-1.0,-0.2,-1.2,-0.1,3.4,0.0,0.8,3.7,2.0)
df_blood=data.frame(drug_A,drug_B)
df_blood

# Variables
x=df_blood$drug_A
y=df_blood$drug_B
new_x=x^2

# Simple Linear Model
model=lm(y ~ x + new_x)
plot(x,y,main="Linear Regression")
lines(x,fitted(model),col="blue")
```

§. 다변수에서의 선형함수 접합 (Fitting a Linear Function of Several Variables)

앞서 논의하였던 다항식 접합과 마찬가지로, 최소제곱법을 이용하여 선형함수의 경우로 일반화할 수 있다. 즉, $y=\beta_0+\beta_1x_1+\beta_2x_2+\cdots+\beta_kx_k$로 일반화할 수 있다. 이를 구하는 방법은, 마찬가지로 점과 임의의 다항식 사이의 제곱합 Q를 최소화하는 계수 값을 찾으면 된다. 이 때 정규방정식은 $k+1$개의 Q에 대한 편미분을 푸는 것으로써, $k+1 \times k+1$ 행렬이 된다. 일련의 과정을 거치면, 최소제곱법을 이용한 선형함수의 접합을 $y=\hat{\beta}_0+\hat{\beta}_1x_1+\hat{\beta}_2x_2+\cdots+\hat{\beta}_kx_k$ 로 얻을 수 있다.

▷ **예제 11.1.5 : Fitting a Linear Function of Two Variables.**

혈압에 관한 이전 예제 11.1.1을 참고하자. 예제 11.1.1의 내용 중, 설명변수인 심박수 x_{i2}를 추가할 것이다. 우리는 반응 y_i를 예측하기 위해서 곡선접합을 사용하여 점들의 추세선이 되는 함수를 구할 것이다. 즉, $y=\beta_0+\beta_1x_1+\beta_2x_2$인 선형함수의 계수들을 구하고자 한다. 다음 표를 참조하자.

Table 11.2 Reactions to two drugs and heart rate

i	x_{i1}	x_{i2}	y_i
1	1.9	66	0.7
2	0.8	62	-1.0
3	1.1	64	-0.2
4	0.1	61	-1.2
5	-0.1	63	-0.1
6	4.4	70	3.4
7	4.6	68	0.0
8	1.6	62	0.8
9	5.5	68	3.7
10	3.4	66	2.0

Table 11.2의 내용을 토대로 선형함수의 계수들을 구하기 위해 최소제곱법을 적용한다. 이 때, 정규방정식은 다음과 같이 구해진다.

$$\begin{aligned}10\beta_0+23.3\beta_1+650\beta_2&=8.1\\23.3\beta_0+90.37\beta_1+1563.6\beta_2&=43.59\\90.37\beta_0+1563.6\beta_1+42334\beta_2&=563.1\end{aligned}$$

이러한 정규방정식을 계수들에 관해 풀면 $\widehat{\beta_0}=-11.4527$, $\widehat{\beta_1}=0.4503$, $\hat{\beta}_2=0.1725$로 구할 수 있다. 따라서 최소제곱 선형함수는 $y=-11.4527+0.4503x_1+0.1725x_2$로 구해진다. R 프로그래밍을 이용하여 구한 결과는 아래에 나타내었다.

R Code in Example 11.1.5

Results

```
> df_blood
   drug_A heart_rate drug_B
1     1.9         66    0.7
2     0.8         62   -1.0
3     1.1         64   -0.2
4     0.1         61   -1.2
5    -0.1         63   -0.1
6     4.4         70    3.4
7     4.6         68    0.0
8     1.6         62    0.8
9     5.5         68    3.7
10    3.4         66    2.0

> model

Call:
lm(formula = y ~ x1 + x2)

Coefficients:
(Intercept)           x1           x2
   -11.4528       0.4503       0.1725
```

R Code

```
# Table
drug_A<-c(1.9,0.8,1.1,0.1,-0.1,4.4,4.6,1.6,5.5,3.4)
heart_rate<-c(66,62,64,61,63,70,68,62,68,66)
drug_B<-c(0.7,-1.0,-0.2,-1.2,-0.1,3.4,0.0,0.8,3.7,2.0)
df_blood=data.frame(drug_A,heart_rate,drug_B)
df_blood

# Variables
x1=df_blood$drug_A
x2=df_blood$heart_rate
y=df_blood$drug_B

# Simple Linear Model
model=lm(y ~ x1 + x2)
```

▲

11.2 회귀(Regression)

§. 회귀함수(Regression functions)

Definition 11.2.1 **반응변수/예측변수/회귀** (Response/Predictor/Regression)
회귀분석과 관련한 통계적 문제에서, 변수 $X_1, ..., X_k$을 **예측변수(Predictor)** 라고 부르며, 이에 대한 확률변수 Y를 **반응변수(Response)** 라고 부른다. $X_1, ..., X_k$의 관측값 $x_1,...,x_k$이 주어진 Y에 대한 조건부 기댓값, 즉 $E(Y|x_1,...,x_k)$을 Y에 대한 **회귀함수(Regression function)** 또는 $X_1, ..., X_k$에서의 Y에 대한 **회귀(Regression)** 이라 부른다.

▷이 장에서는 $E(Y|x_1,...,x_k)=\beta_0+\beta_1x_1+\beta_2x_2+\cdots+\beta_kx_k$로 생각하자. 이 때, 이 선형함수의 각 계수를 **회귀계수(regression coefficients)**라고 부르며, 이들은 알려져 있지 않은 것으로 간주한다.

▷ **예제 11.2.1 : Pressure and the Boiling Point of Water.**
1857년, Forbes는 고도(altitude)를 추정하기 위한 방법을 얻기 위한 실험의 결과를 보고하였다. 그러한 고도를 측정하는 방법은 기압계 상의 압력을 측정하는 것으로 얻을 수 있으나, Forbes가 살던 시기에는 높은 고도로 기압계를 가져가기 어려웠던 상황이었다. 그러나 현대에는 많은 여행객들이 산을 오를 때 가지고 다니는 온도계를 소지하고 물의 끓는점을 측정하는 것으로 고도를 손쉽게 구할 수 있다. 다음 Table 11.5 에서 이러한 끓는점과 압력을 나타내었다.

Table 11.5 Boiling point of water in degrees Fahrenheit and atmospheric pressure in inches of mercury from Forbes' experiments.

Boiling Point	Pressure
194.5	20.79
194.3	20.79
197.9	22.40
198.4	22.67
199.4	23.15
199.9	23.35
200.9	23.89
201.1	23.99
201.4	24.02
201.3	24.01
203.6	25.14
204.6	26.57
209.5	28.49
208.6	27.76
210.7	29.04
211.9	29.88
212.2	30.06

우린 여기서 끓는점과 압력 사이의 선형관계를 접합하기 위해 최소제곱법을 이용할 수 있다. y_i를 Forbes가 관측한 관측값들 중 어느 값이라 하고 x_i를 끓는점이라 하자. 이 때, $i=1,...,17$이다. Table 11.5의 데이터들을 이용하면, 우린 최소제곱선을 계산할 수 있다. 그러한 최소제곱선을 계산하면 절편과 기울기는 각각 $\hat{\beta}_0 = -81.049$와 $\hat{\beta}_1 = 0.5228$로 구해진다. 따라서 최소제곱선 $y = -81.049 + 0.5228x$로 구할 수 있다. 물론, 우리는 이러한 최소제곱선이 끓는점과 압력 사이의 관계를 자세히 제시할 수 없다는 것을 알고 있다. 만약 우리가 물의 끓는점 x를 계속해서 얻을 수 있고 알려지지 않은 압력 Y에 대한 조건부 분포를 계산하길 원한다면, 그러한 Y를 계산할 수 있게 해줄 통계적 모델의 존재 여부가 궁금할 것이다.

▲

§. 단순선형회귀(Simple Linear Regression)

단순선형회귀에서는 회귀식이 단일 변수인 X에 대한 회귀 Y에 관한 식으로 간주한다. 우리는 각 설명변수가 $X=x$로, 확률변수 Y는 $Y=\beta_0+\beta_1 x+\epsilon$, where random variable $\epsilon \sim N(0,\sigma^2)$ 로 가정한다. 이 장에서 우린 각 점 (x_1, Y_1), ..., (x_n, Y_n)에 대한 회귀 문제를 다루게 될 것이다. 다음의 가정들은 단순선형회귀를 포함한 여러 개의 예측변수들(설명변수)에서의 회귀를 위해 반드시 필요한 가정들이므로 잘 숙지해야 한다.

Assumption 11.2.1 예측변수가 알려져 있다 (Predictor is known)
값 $x_1, ..., x_n$이 미리 알려져 있거나 $(Y_1, ..., Y_n)$의 결합분포를 계산하기 전에 확률변수 $X_1, ..., X_n$의 관측 값이어야 한다.

Assumption 11.2.2 정규성 (Normality)
$i=1,...,n$에 대하여, $x_1,...,x_n$이 주어진 Y_i의 조건부 분포가 정규분포여야 한다.

Assumption 11.2.3 선형성을 갖는 평균 (Linear Mean)
$i=1,...,n$에 대하여, $E(Y_i|x_1,...,x_n)$이 $\beta_0+\beta_1 x$ 의 형태인 모수 β_0과 β_1이 존재함을 말한다.

Assumption 11.2.4 등분산성 (Homoscedasticity)
$i=1,...,n$에 대하여, $Var(Y_i|x_1,...,x_n)=\sigma^2$인 모수 σ^2가 존재함을 말한다. 여기서 다른 분산을 갖는 확률변수는 **이분산성(Heteroscedasticity)**를 갖는다고 말한다.

Assumption 11.2.5 독립성 (Independence)
관측 값 $x_1,...,x_n$가 주어진 확률변수 $Y_1,...,Y_n$이 독립적임을 일컫는다.

앞서 설명한 가정 11.2.1-11.2.5는 $\vec{x}=(x_1,\ ...,x_n)$이 주어진 조건부 결합 분포 $Y_1,...,Y_n$을 특정하고, 모수(매개변수) β_0, β_1, σ^2를 특정한다. 특히 조건부 결합 p.d.f.인 $Y_1,...,Y_n$는 다음과 같이 표현할 수 있다.

$$f_n(\vec{y}\,|\,\vec{x},\beta_0,\beta_1,\sigma^2)=\frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i-\beta_0-\beta_1x_i)^2\right]$$

우린 이러한 β_0,β_1,σ^2의 최대우도추정량을 찾을 것이다.

Theorem 11.2.1 단순선형회귀의 최대우도추정량 (Simple Linear Regression M.L.E.'s)
Assumption 11.2.1-11.2.5를 만족한다고 가정하자. β_0, β_1의 최대우도추정량은 최소제곱추정량이고, σ^2의 최대우도추정량은 다음과 같다.

$$\hat{\sigma}^2=\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{\beta}_0-\hat{\beta}_1x_i)^2$$

§. 최소제곱추정량의 분포 (The Distribution of the Least-Squares Estimators)

Theorem 11.2.2 최소제곱추정량의 분포 (Distribution of Least-Squares Estimators)
Assumption 11.2.1-11.2.5를 만족한다고 가정하자. 그러면 최소제곱추정량 $\hat{\beta}_1$의 분포는 평균 β_1과 분산 σ^2/s_x^2를 갖는 정규분포를 따른다. 또한, $\hat{\beta}_0$의 분포는 평균 β_0과 분산 $\sigma^2\left(\frac{1}{n}+\frac{\bar{x}^2}{s_x^2}\right)$ 을 갖는 정규분포를 따른다. 마지막으로, 두 최소제곱추정량의 공분산(covariance)은 다음과 같다.

$$Cov(\hat{\beta}_0,\hat{\beta}_1)=\frac{\bar{x}\,\sigma^2}{s_x^2}$$

이 정리에서의 모든 분포에 관한 설명은 확률변수 X_i에 대한 조건부로 $X_i=x_i$을 가짐을 유의한다.

▷ **예제 11.2.2 : Pressure and the Boiling Point of Water.**
이전 예제 11.2.1을 참고하자. 우린 앞서 구하였던 최소제곱추정량들의 분산과 공분산을 계산하고자 한다. 우선 평균 온도는 $\bar{x}=202.95$, $s_x^2=530.78$, 전체 관측 값의 개수는 $n=17$로 구할 수 있다. 우린 σ^2를 알지 못하므로, 이를 제하고 정리 11.2.2를 이용하여 분산과 공분산을 계산해본다. 그 결과는 아래와 같다.

$$Var(\hat{\beta}_1)=\frac{\sigma^2}{530.78}=0.00188\sigma^2,\quad Var(\hat{\beta}_0)=\sigma^2\left(\frac{1}{17}+\frac{202.95^2}{530.78}\right)=77.66\sigma^2,$$

$$Cov(\hat{\beta}_0,\hat{\beta}_1)=-\frac{202.95\,\sigma^2}{530.78}=0.382\sigma^2.$$

이 결과들을 잘 살펴보면, β_0보다 β_1의 추정치가 더 자세한 값임을 예상할 수 있다.

▲

▷ **예제 11.2.3 : The Variance of a Linear Combination.**

회귀분석과 관련한 통계학적 문제에서, 최소제곱추정량의 선형결합의 분산을 자주 계산할 필요가 있다. 그러한 하나의 예로써 '예측'에 관한 문제가 있다. 우리는 $T = c_0\hat{\beta}_0 + c_1\hat{\beta}_1 + c_*$의 분산을 계산하고 싶다고 가정하자. T의 분산은 다음과 같이 $Var(\hat{\beta}_0)$, $Var(\hat{\beta}_1)$, $Cov(\hat{\beta}_0, \hat{\beta}_1)$의 값으로 대체함으로써 찾아질 수 있다.

$$Var(T) = c_0^2 Var(\hat{\beta}_0) + c_1^2 Var(\hat{\beta}_1) + 2c_0c_1 Cov(\hat{\beta}_0, \hat{\beta}_1)$$

이는 앞서 정의하였던 $Var(\hat{\beta}_0)$, $Var(\hat{\beta}_1)$, $Cov(\hat{\beta}_0, \hat{\beta}_1)$의 정의를 대입하여 정리하면 다음과 같이 쓸 수 있다.

$$Var(T) = \sigma^2\left(\frac{c_0^2}{n} + \frac{(c_0\overline{x} - c_1)^2}{s_x^2}\right)$$

예제 11.2.2의 특별한 경우로, 우리는 $c_0 = 0$, $c_1 = 200$으로 두었다. 따라서 $200\hat{\beta}_1$의 분산은 $200^2\sigma^2/s_x^2 = 75.36\sigma^2$로 구해진다. 이는 $\hat{\beta}_0$의 분산 값 $77.66\sigma^2$와 매우 가까운 값으로 구해짐을 확인할 수 있다.

▲

▷ **예제 11.2.5 : Pressure and the Boiling Point of Water.**

예제 11.2.4에서, 우리는 물의 끓는 점이 201.5도일 때의 기압을 예측하길 원했다. 최소제곱선은 $y = -81.049 + 0.5228x$로 구해지고, $\hat{\sigma}^2 = 0.0478$로 구하였다. 기압 Y의 예측값의 M.S.E.(Mean Squared Error)는 다음으로부터 얻어진다.

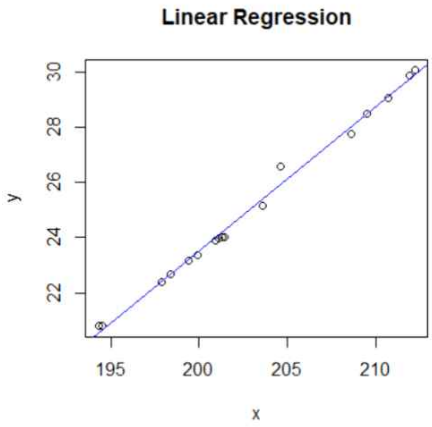
$$E[(\hat{Y} - Y)^2] = \sigma^2\left[1 + \frac{1}{17} + \frac{(201.5 - 202.95)^2}{530.78}\right] = 1.0628\sigma^2$$

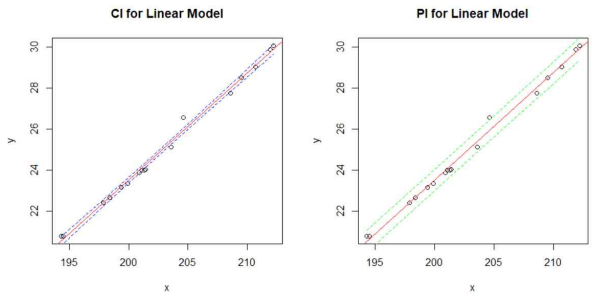
또한, 예측 $\hat{Y}$의 값은 $\hat{Y} = -81.06 + 0.5229 \times 201.5 = 24.30$으로 관측된다. M.S.E.의 값 $1.0628\sigma^2$는 다음과 같이 해석될 수 있다. : 만약 우리가 β_0과 β_1의 값을 알고 있고, Y를 예측하고자 한다면, M.S.E.는 $Var(Y) = \sigma^2$가 될 것이다. β_0과 β_1을 추정할 때 우리가 가져야할 것은 M.S.E.에서 오직 $0.0628\sigma^2$ 뿐이다.

아래에 R프로그래밍으로 이러한 예측값을 구하고 회귀선을 구한 코드를 제시하였다.

R Code in Example 11.2.5

Results






```
> # Prediction of specific value
> Linefunc(201.5)
[1] 24.29909
```

R Code

```
# Table
Boil<-c(194.5,194.3,197.9,198.4,199.4,199.9,200
.9,201.1,201.4,201.3,203.6,204.6,209.5,208.6,210
.7,211.9,212.2)
Pres<-c(20.79,20.79,22.40,22.67,23.15,23.35,23.
89,23.99,24.02,24.01,25.14,26.57,28.49,27.76,29.
04,29.88,30.06)
df_water=data.frame(Boil,Pres)
df_water
# Variables
x=df_water$Boil
y=df_water$Pres
# Simple Linear Model
model=lm(y ~ x)
plot(x,y,main="Linear Regression")
abline(model,col="blue")
#Linear function
Linefunc<-function(x){
-81.0637271+0.5228924*x}
# Prediction of specific value
Linefunc(201.5)
par(mfrow=c(1,2))
# 95% Confidence Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),
interval="confidence",level=0.95)
plot(x,y,main="CI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="blue",lty=2)
lines(newx,CI[,3],col="blue",lty=2)
# 95% Prediction Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),
interval="prediction",level=0.95)
plot(x,y,main="PI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="green",lty=2)
lines(newx,CI[,3],col="green",lty=2)
```

▲

11.3 단순선형회귀의 통계적 추론(Statistical Inference in Simple Linear Regression)

§. 추정량의 결합분포(Joint Distribution of the Estimators)

▷ **예제 11.3.1 : Pressure and the Boiling Point of Water.**

예제 11.2.4에서, 끓는 점이 201.5일 때의 기압을 단순선형회귀를 이용하여 예측해 보았다. 여행객들이 201.5도에서 기압이 24.5인지 그 여부를 알고 싶어 한다고 가정하자. 즉, 다음과 같은 가설을 검정하고 싶어 한다.

$$H_0 : \beta_0 + 201.5\beta_1 = 24.5$$

혹은 그 대신에, $\beta_0 + 201.5\beta_1$의 구간 추정치를 알고 싶다고 하자. 이러한 추론들은 회귀 모형의 모든 모수들 $(\beta_0, \beta_1, \sigma^2)$의 추정량의 결합분포를 찾는 것으로 가능하다.

▲

Theorem 11.3.1

확률변수 $Y_1, \ldots, Y_n$이 독립이고, 각각이 모두 분산 σ^2를 갖는 정규분포를 따른다고 가정하자. $Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$, $Z = \begin{bmatrix} Z_1 \\ \vdots \\ Z_n \end{bmatrix}$인 확률변수의 벡터라고 하자. 만약 A가 $n \times n$ 직교행렬이고 $Z = AY$라 하면, 확률변수 $Z_1, \ldots, Z_n$ 또한 독립이고 각각이 모두 분산 σ^2를 갖는 정규분포를 따른다.

Theorem 11.3.2

단순선형회귀의 문제에서, $(\hat{\beta}_0, \hat{\beta}_1)$의 결합분포는 각각이 다음의 평균과 분산을 갖는 이변수(bivariate) 정규분포를 갖는다.

$$\hat{\beta}_1 \text{ 의 분포 : 평균 } \beta_1 \text{ , 분산 } \sigma^2/s_x^2$$

$$\hat{\beta}_0 \text{ 의 분포 : 평균 } \beta_0 \text{과 분산 } \sigma^2\left(\frac{1}{n} + \frac{\overline{x}^2}{s_x^2}\right)$$

또한, $n \geq 3$이면, $\hat{\sigma}^2$는 $(\hat{\beta}_0, \hat{\beta}_1)$와 독립적이고 $n\hat{\sigma}^2/\sigma^2$는 자유도가 $n-2$인 χ^2 분포를 갖는다.

§. 회귀계수에 대한 가설검정(Tests of Hypotheses about the Regression Coefficients)

β_0과 β_1의 선형결합에 대한 가설 검정

c_0, c_1, c_*은 c_0와 c_1이 0이 아닌 특정된 값들이라 하자. 우리는 다음과 같은 가설을 검정하고자 한다.

$$H_0 : c_0\beta_0 + c_1\beta_1 = c_*$$
$$H_1 : c_0\beta_0 + c_1\beta_1 \neq c_*$$

우리는 확률변수 $c_0\hat{\beta}_0 + c_1\hat{\beta}_1$과 σ'를 기반으로 이러한 가설의 검정을 유도하고자 한다.

▷ **단측 가설검정(One-Sided Test)**의 경우는 앞서 해왔던 여러 단측 검정과 동일하게 설정할 수 있다. 이 때, 단측 검정의 귀무가설의 부등호와 반대인 $U_{01} \geq T_{n-2}^{-1}(1-\alpha_0)$ 또는 $U_{01} \leq T_{n-2}^{-1}(1-\alpha_0)$인 경우 귀무가설을 기각한다.

Theorem 11.3.3

각 $0 < \alpha_0 < 1$ 에 대하여, 위와 같은 가설에 대한 수준 α_0 검정은 다음과 같은 조건일 때 귀무가설 H_0을 기각한다.

$$|U_{01}| \geq T_{n-2}^{-1}(1-\alpha_0/2) \text{ where } U_{01} = \left[\frac{c_0^2}{n} + \frac{(c_0\overline{x} - c_1)^2}{s_x^2}\right]^{-1/2}\left(\frac{c_0\hat{\beta}_0 + c_1\hat{\beta}_1 - c_*}{\sigma'}\right).$$

T_{n-2}^{-1} 는 $n-2$ 자유도인 t 분포의 분계함수(quantile function) 이다.

β_0에 대한 가설검정

β_0^*은 특정된 값으로 $-\infty < \beta_0^* < \infty$인 대소 관계를 갖는다고 하고, 다음과 같은 가설을 검정하고자 한다.

$$H_0 : \beta_0 = \beta_0^*$$
$$H_1 : \beta_0 \neq \beta_0^*$$

$c_0 = 1, c_1 = 0, c_* = \beta_0^*$로 두면, $U_0 = \dfrac{\hat{\beta}_0 - \beta_0^*}{\sigma'\left[\dfrac{1}{n} + \dfrac{\overline{x}^2}{s_x^2}\right]^{1/2}}$ 이고 귀무가설이 참일 때 마찬가지로 자유도 $n-2$인 t 분포를 갖는다. 물론, p-값을 구하는 것은 이전의 t 분포와 마찬가지로 구하면 된다.

만약 회귀 계수가 아니라 회귀선 자체가 원점을 지난다 vs 원점을 지나지 않는다 에 대한 가설 검정을 실시하려면 다음과 같이 가설을 설정하면 된다.

$$H_0 : \beta_0 = \beta_0^* = 0$$
$$H_1 : \beta_0 \neq \beta_0^* = 0$$

β_1에 대한 가설검정

β_1^*은 특정된 값으로 $-\infty < \beta_1^* < \infty$인 대소 관계를 갖는다고 하고, 다음과 같은 가설을 검정하고자 한다.

$$H_0 : \beta_1 = \beta_1^*$$
$$H_1 : \beta_1 \neq \beta_1^*$$

$c_0 = 0, c_1 = 1, c_* = \beta_0^*$로 두면, $U_1 = s_x \dfrac{\hat{\beta}_1 - \beta_1^*}{\sigma'}$ 이고 귀무가설이 참일 때 마찬가지로 자유도 $n-2$인 t 분포를 갖는다.

만약 회귀 계수가 아니라 회귀선의 X와 Y가 실제로 상관관계가 없음을 검정하고 싶으면 다음과 같이 가설을 설정하면 된다.

$$H_0 : \beta_1 = \beta_1^* = 0$$
$$H_0 : \beta_1 \neq \beta_1^* = 0$$

회귀선에 대한 가설검정

회귀선 $y = \beta_0 + \beta_1 x$이 특정한 점 (x^*, y^*), $x^* \neq 0$을 지나는지 여부에 대한 가설검정을 하고자 한다. 그러면 다음과 같이 가설을 설정하면 된다.

$$H_0 : \beta_0 + \beta_1 x^* = y^*$$
$$H_1 : \beta_0 + \beta_1 x^* \neq y^*$$

이 가설은 $c_0 = 1$, $c_1 = x^*$, $c_* = y^*$로 두고 U_{01}이 마찬가지로 $n-2$ 자유도인 t 분포를 따른다.

▷ **예제 11.3.3 : Pressure and the Boiling Point of Water.**

이제, 앞선 예제 11.3.1에서 검정하고자 했던 다음 가설을 검정해보자. 그 가설은 다음과 같이 세울 수 있다.

$$H_0 : \beta_0 + 201.5\beta_1 = 24.5$$
$$H_1 : \beta_0 + 201.5\beta_1 \neq 24.5$$

통계량을 $c_0 = 1$, $c_1 = 201.5$로 두어 구한다. 앞서 구하였던 최소제곱추정치는 $\hat{\beta}_0 = -81.049$와 $\hat{\beta}_1 = 0.5228$ 이다. 또한 $n = 17$, $s_x^2 = 530.78$, $\overline{x} = 202.95$, $\sigma' = 0.2328$로 구하였다. 따라서 $U_{01} = -0.2204$로 구할 수 있다. 만약 H_0이 참이면 U_{01}은 $n-2 = 15$ 자유도를 갖는 t 분포를 따른다. 검정통계량 -0.2204에 대응하는 p-값은 0.8285로 구할 수 있다. 따라서 귀무가설은 유의수준 α_0에 대하여 $\alpha_0 \geq 0.8285$ 일 때 기각한다.

▲

§. 신뢰 구간과 예측 구간(Confidence Intervals & Prediction Intervals)

Theorem 11.3.5

c_0, c_1 이 0이 아닌 상수라고 하자. 두 확률변수 β_0, β_1 사이의 열린구간은 다음과 같다.

$$c_0\hat{\beta}_0 + c_1\hat{\beta}_1 \pm \sigma' \left[\frac{c_0^2}{n} + \frac{(c_0\bar{x} - c_1)^2}{s_x^2} \right]^{1/2} T_{n-2}^{-1}\left(1 - \frac{\alpha_0}{2}\right)$$

이는 $1-\alpha_0$ **계수의** $c_0\beta_0 + c_1\beta_1$**에 대한 신뢰구간(Coefficient Interval)**이라 부른다.

Theorem 11.3.6 & Definition 11.3.1

단순선형회귀의 문제에서, Y를 Y_1, ..., Y_n이 독립일 때 설명변수 x로 인해 구해진 새로운 관측값이라 하자. 그리고 $\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x$라 하자. 그러면 Y가 다음 두 확률변수 사이에 있을 확률은 $1-\alpha_0$이다. 그러한 두 확률변수는 다음과 같다.

$$\hat{Y} \pm T_{n-2}^{-1}\left(1 - \frac{\alpha_0}{2}\right)\sigma' \left[1 + \frac{1}{n} + \frac{(x-\bar{x})^2}{s_x^2} \right]^{1/2}$$

여기서 이는 $1-\alpha_0$ **계수의** Y **에 대한 예측구간(Prediction Interval)**이라 부른다.

§. 잔차 분석(The Analysis of Residuals)

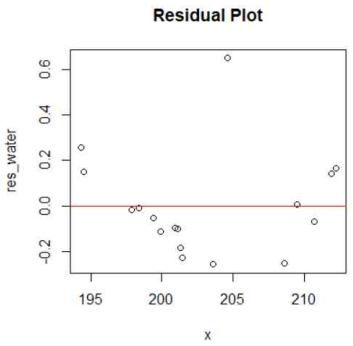
Definition 11.3.2 잔차/적합값

$i = 1, \ldots, n$에 대하여, $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$의 관측값들을 **적합값(Fitted value)** 이라고 부르고, $e_i = y_i - \hat{y}_i$를 **잔차(Residuals)** 라고 부른다.

▷ **예제 11.3.6 : Pressure and the Boiling Point of Water.**

이전 같은 주제의 예제에서 소개한 Table을 참조하여, R 프로그래밍으로 구한 잔차 그래프로 대신한다.

Results

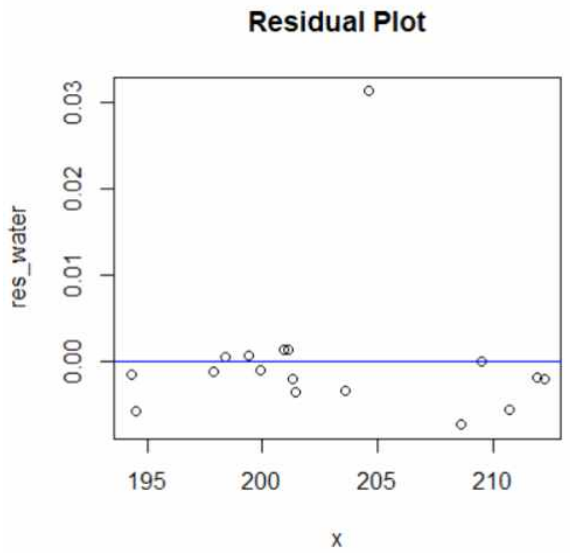


R Code

```
# Residual Plot
res_water=resid(model)
plot(x,res_water,main="Residual Plot")
abline(0,0,col="red")
```

잔차 그래프를 보면, 점들이 이상점(outlier)을 제외하고 U자 형태의 패턴으로 뿌려져 있는 것을 확인할 수 있다. 이는 구한 회귀선이 선형회귀보단 곡선이 더 적합함을 암시한다. 그러나 이는 수치상에서 나타나는 현상으로 실제로도 선형이 적합하지 않은지는 정확히 알 수 없다. 따라서 데이터를 로그를 취한 로그 스케일로 다시 분석해보는 것이 필요하다. 여기서 로그 스케일을 사용하는 이유는 나타난 실제 값들이 아닌 데이터의 본질을 파악하기 위함이다. 예를 들어 1000일 때 100000을 반환하는 함수가 있다고 하자. 그러면 이는 10일 때 1000을 반환하는 함수랑 같다. 그러나 수치상으로는 첫 번째가 훨씬 큰 값을 반환하므로 어떤 오류가 있다고 생각할 수 있다. 로그 스케일을 적용하는 것은 이러한 일들을 테크니컬하게 분별하기 위한 일종의 도구이다. 로그 스케일을 씌워서 만든 잔차 그래프는 아래와 같이 구해진다.

Results




```
> model

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
    -0.97087      0.02062
```

R Code

```
# Table
Boil<-c(194.5,194.3,197.9,198.4,199.4,199.9,200
.9,201.1,201.4,201.3,203.6,204.6,209.5,208.6,210
.7,211.9,212.2)
Pres<-c(20.79,20.79,22.40,22.67,23.15,23.35,23.
89,23.99,24.02,24.01,25.14,26.57,28.49,27.76,29.
04,29.88,30.06)
Pres_lo<-log(Pres)
df_water=data.frame(Boil,Pres_lo)
df_water

# Variables
x=df_water$Boil
y=df_water$Pres_lo

# Simple Linear Model_log
model=lm(y ~ x)

# Residual Plot
res_water=resid(model)
plot(x,res_water,main="Residual Plot")
abline(0,0,col="blue")
```

▲

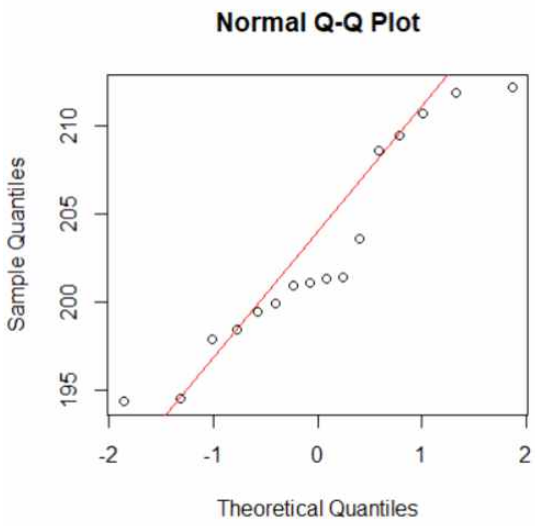
▷ **예제 11.3.7 : Pressure and the Boiling Point of Water.**

Normal Quantile Plot, 줄여서 **정규 Q-Q 플롯**에 대하여 잠시 소개하도록 하자. 자세한 배경은 교재를 참고한다. Q-Q (Quantile-Quantile) 플롯은 데이터가 특정 분포를 따르는지를 시각적으로 검토하는 방법이다. 예를 들어, 처음에 주어진 데이터가 정규분포를 따르는지 살펴보고 싶다고 가정한다. $X \sim N(\mu,\sigma^2)$이라면 표준화 하였을 때 $Z=(X-\mu)/\sigma \sim N(0,1)$임을 안다. 여기서 X를 Z에 대한 식으로 표현하면, $X=\mu+\sigma Z$ 이다. 따라서 X가 정규분포를 따를 때 (X,Z)를 좌표평면에 표시한다면, 식 $X=\mu+\sigma Z$은 직선의 형태로 평면상에 나타나게 될 것이다.

한편으로는, 이는 또한 잔차의 정규성을 확인할 때 사용하는 플롯으로, 이는 단순선형회귀의 여러 Assumption 중 하나인 오차가 $N(0,\sigma^2)$를 따른다는 것을 대략적으로 확인하는 과정이다. 이는 그래프 상에 직선과 점들이 얼마나 가까운지에 따라 데이터가 정규성을 띄는지 여부를 확인할 수 있다. 직선은 이론적으로 정규분포일 때 만들어져야 할 이상적인 직선이고($y=x$), 점은 실제 데이터가 따르는 위치이다. 이러한 점들이 직선에 가까울수록 데이터는 더욱 정규성을 띈다고 볼 수 있다.

이제 앞선 예제의 로그 스케일 데이터를 그대로 이용하여 Q-Q 플롯을 확인해보자. 통계 소프트웨어를 이용한다. 예제에서는 이상점이 되는 데이터 값(Table의 12번째 값)을 제거한 후 출력하였다.

Results



R Code

```
# Table
Boil<-c(194.5,194.3,197.9,198.4,199.4,199.9,200
.9,201.1,201.4,201.3,203.6,209.5,208.6,210.7,211
.9,212.2)
Pres<-c(20.79,20.79,22.40,22.67,23.15,23.35,23.
89,23.99,24.02,24.01,25.14,28.49,27.76,29.04,29.
88,30.06)
Pres_lo<-log(Pres)
df_water=data.frame(Boil,Pres_lo)
df_water

# Variables
x=df_water$Boil

# Q-Q Plot
qqnorm(x)
qqline(x,distribution=qnorm,col="red")
```

▲

11.5 일반선형모형과 다중회귀(The General Linear Model and Multiple Regression)

§. 일반선형모형(The general Linear Model)

이 장에서 다룰 일반선형모형은 단순선형회귀모형에서 확장된 형태로, 단순선형회귀모형의 형태가 여러개의 설명변수를 갖는 것을 일컫는다. 우리는 이러한 경우의 회귀분석을 실시하기 전에, 이러한 다변수의 상황에서 벡터와 행렬을 이용한 일반화된 식을 다룰 것이다.

▷ 일반선형모형에서의 가정은 기존의 단순선형회귀와 같다. 다만 행렬의 관점에서 서술한 것뿐이다.

Assumption 11.2.1 예측변수가 알려져 있다 (Predictor is known)
값 $z_1, \ldots, z_n$이 미리 알려져 있거나 $(Y_1, \ldots, Y_n)$의 결합분포를 계산하기 전에 확률변수 $Z_1, \ldots, Z_n$의 관측 값이어야 한다.

Assumption 11.2.2 정규성 (Normality)
$i = 1, \ldots, n$에 대하여, $z_1, \ldots, z_n$이 주어진 Y_i의 조건부 분포가 정규분포여야 한다.

Assumption 11.2.3 선형성을 갖는 평균 (Linear Mean)
$i = 1, \ldots, n$에 대하여, $E(Y_i | z_1, \ldots, z_n)$이 $z_{i0}\beta_0 + z_{i1}\beta_1 + \ldots + z_{ip-1}\beta_{p-1}$ 의 형태인 모수 $\beta = (\beta_0, \ldots, \beta_{p-1})$이 존재함을 말한다.

Assumption 11.2.4 등분산성 (Homoscedasticity)
$i = 1, \ldots, n$에 대하여, $Var(Y_i | z_1, \ldots, z_n) = \sigma^2$인 모수 σ^2가 존재함을 말한다. 여기서 다른 분산을 갖는 확률변수는 **이분산성(Heteroscedasticity)**를 갖는다고 말한다.

Assumption 11.2.5 독립성 (Independence)
관측 값 $z_1, \ldots, z_n$가 주어진 확률변수 $Y_1, \ldots, Y_n$이 독립적임을 일컫는다.

Definition 11.5.1 일반선형모형 (General Linear Model)
위의 Assumption을 만족하는 관측 값 $z_1, \ldots, z_n$가 주어진 확률변수 $Y_1, \ldots, Y_n$의 통계적 모형을 말한다.

▷ **예제 11.5.1-2 : Unemployment in the 1950s.**

Table 11.12에서 제공되는 1950년부터 1959년까지 10년간의 실업률과 생산 지수에 관한 데이터가 주어졌다. 이 Table은 하나 이상의 설명변수를 갖고 있다. 이 때 우리는 어떻게 회귀모형을 구할 수 있는지에 주목하고자 한다.

Table 11.12 Unemployment data for Example 11.5.1

Unemployment	Index of production	Year
3.1	113	1950
1.9	123	1951
1.7	127	1952
1.6	138	1953
3.2	130	1954
2.7	146	1955
2.6	151	1956
2.9	152	1957
4.7	141	1958
3.8	159	1959

우린 실업률을 예측하고 싶기 때문에 생산 지수와 년도를 설명변수로 둘 것이다. 즉, X_1을 생산 지수로, X_2를 년도로 설정할 것이다.

▲

일반선형모형에서의 최대우도추정량 (M.L.E.)

단순선형회귀에서의 최대우도추정량과 마찬가지 방법으로 구한다. 그러한 결과는 다음과 같다.

$$S^2 = \sum_{i=1}^{n} (Y_i - z_{i0}\hat{\beta}_0 - \cdots - z_{ip-1}\hat{\beta}_{p-1})^2$$

σ^2의 최대우도추정량(M.L.E.)은 $\hat{\sigma}^2 = \frac{S^2}{n}$ 이고,

σ의 불편추정량(unbiased estimator)은 $\sigma' = \left(\frac{S^2}{n-p}\right)^{1/2}$ 이다.

§. 추정량의 양형태적 표현(Explicit Form of the Estimators)

추정량들을 벡터와 행렬을 이용하여 표현하는 것을 주로 양형태(Explicit Form)이라 하고, 양형태적 표현을 사용함으로써 표기를 간편하게 줄이고, 행렬대수적인 부분을 편리하게 할 수 있다. 일반선형모형에서, 그러한 표현은 다음과 같이 정의된다.

Definition 11.5.2 계획행렬

일반선형모형에서 $n \times p$ 행렬 $Z = \begin{bmatrix} z_{10} \cdots z_{1p-1} \\ z_{20} \cdots z_{2p-1} \\ \vdots \ddots \vdots \\ z_{n0} \cdots z_{np-1} \end{bmatrix}$ 를 모형의 **계획행렬(design matrix)**이라 부른다.

▷ '계획행렬' 이란 이름은 z_{ij}가 잘 설계된 실험을 만들기 위한 실험자에 의해 선택될 수 있음을 말한다.

Definition 일반선형모형의 요인들

$Y_1, \ldots, Y_n$의 관측값들과 회귀모수들, 회귀계수추정량들은 다음과 같은 벡터 형태로 나타낼 수 있다.

$$\vec{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \vec{\beta} = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_{p-1} \end{bmatrix}, \quad \vec{\hat{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \vdots \\ \hat{\beta}_{p-1} \end{bmatrix}$$

임의의 벡터 또는 행렬의 전치(transpose)는 v일 때 v'로 표기한다.

Theorem 11.5.1 일반선형모형 추정량

$\vec{\beta}$의 최소제곱추정량(또는 최대우도추정량)은 $\vec{\hat{\beta}} = (Z'Z)^{-1}Z'Y$ 이다.

▷ **예제 11.5.3 : Unemployment in the 1950s.**

Table 11.12를 참조하면, 계획행렬 Z는 $n \times p = 10 \times 3$ 행렬로 주어지고, 행렬의 첫 번째 열은 모두 1로 채우도록 한다. 그리고 두 번째 열은 Table 11.12의 두 번째 열로 놓는다. 마지막으로, 수치적인 문제가 발생하는 것을 피하기 위해, 계획행렬의 세 번째 열은 Table의 세 번째 열에 1949를 빼준 값으로 한다. 즉 다음과 같이 계획행렬을 만들고자 한다. 이 때 $\vec{y}$ 또한 Table의 첫 번째 열로 둔다.

$$Z = \begin{pmatrix} 1 & 113 & 1950-1949 \\ \vdots & \vdots & \vdots \\ 1 & 159 & 1959-1949 \end{pmatrix} = \begin{pmatrix} 1 & 113 & 1 \\ \vdots & \vdots & \vdots \\ 1 & 159 & 10 \end{pmatrix}, \quad \vec{y} = \begin{pmatrix} 3.1 \\ \vdots \\ 3.8 \end{pmatrix}$$

이제 $(Z'Z)^{-1}$과 $Z'\vec{y}$를 계산하면 $(Z'Z)^{-1} = \begin{pmatrix} 38.35 & -0.3323 & 1.383 \\ -0.3323 & 2.915 \times 10^{-3} & -0.01272 \\ 1.383 & -0.01272 & 0.06762 \end{pmatrix}$, $Z'\vec{y} = \begin{pmatrix} 28.2 \\ 3931 \\ 144.1 \end{pmatrix}$ 로 얻을 수 있고, 정리 11.5.1의 식을 이용하여 추정량을 구하면 $\hat{\beta} = \begin{pmatrix} 13.45 \\ -0.1033 \\ 0.6594 \end{pmatrix}$ 로 얻을 수 있다.

▲

§. 평균 벡터와 공분산 행렬 (Mean vector and Covariance Matrix)

이 장에선 추정량 벡터의 각 성분 $\hat{\beta}_0, ..., \hat{\beta}_{p-1}$ 의 공분산, 분산, 평균을 유도하고자 한다. $\overrightarrow{Y}$는 n차원 확률벡터로써, 성분을 $Y_1, ..., Y_n$으로 갖는다고 가정하자. 이 확률벡터의 기댓값 $E(\overrightarrow{Y})$는 마찬가지로 $E(Y_1), ..., E(Y_n)$을 성분으로 가지는 n차원 벡터로 정의된다.

Definition 11.5.3 **평균벡터/공분산 행렬**
$\overrightarrow{Y}$가 확률벡터일 때, $E(\overrightarrow{Y})$를 $\overrightarrow{Y}$의 **평균벡터(mean vector)** 라 부르고, $n \times n$ 행렬인 **공분산 행렬(covariance matrix)** 은 다음과 같이 정의된다.

$$Cov(\overrightarrow{Y}) = \begin{bmatrix} \sigma_{11} \cdots \sigma_{1n} \\ \vdots \ \ddots \ \vdots \\ \sigma_{n1} \cdots \sigma_{nn} \end{bmatrix} \text{ if } Cov(Y_i, Y_j) = \sigma_{ij}$$

▷ 공분산 행렬은 대각원소에 대해서 $Var(Y_i) = Cov(Y_i, Y_i) = \sigma_{ii}$ 이고, $Cov(Y_i, Y_j) = Cov(Y_j, Y_i)$ 이므로, $\sigma_{ij} = \sigma_{ji}$ 을 만족한다. 따라서 공분산 행렬은 대칭(symmetric)행렬이 됨을 확인할 수 있다.

Theorem 11.5.2 n차원 확률벡터 $\overrightarrow{Y}$ 에 대하여 $E(\overrightarrow{Y})$와 $Cov(\overrightarrow{Y})$가 존재한다고 가정하자, 그리고 A가 성분이 상수인 $p \times n$ 행렬이고 $\overrightarrow{W}$가 $\overrightarrow{W} = A\overrightarrow{Y}$로 정의된 p차원 확률벡터라고 하자. 그러면 $E(\overrightarrow{W}) = AE(\overrightarrow{Y})$이고 $Cov(\overrightarrow{W}) = ACov(\overrightarrow{Y})A'$이다.

▷이 정리를 이용하여 각 추정량들의 평균과 분산, 공분산이 얻어진다.

Theorem 11.5.3 일반선형모형에서, $E(\overrightarrow{\hat{\beta}}) = \overrightarrow{\beta}$, $Cov(\overrightarrow{\hat{\beta}}) = \sigma^2(Z'Z)^{-1}$ 이다.

§. 추정량의 결합분포 (The Joint Distribution of the Estimators)

Corollary 11.5.1 $p \times p$ 대칭행렬 $(Z'Z)^{-1}$의 성분을 다음과 같이 두자.

$$(Z'Z)^{-1} = \begin{bmatrix} \zeta_{00} & \cdots & \zeta_{0p-1} \\ \vdots & \ddots & \vdots \\ \zeta_{p-10} & \cdots & \zeta_{p-1p-1} \end{bmatrix}$$

$j = 0, ..., p-1$에 대하여, 추정량 $\hat{\beta}_j$는 $\hat{\beta}_j \sim N(\beta_j, \zeta_{jj}\sigma^2)$를 만족한다. 또한 $i \neq j$에 대하여 추정량의 공분산은 $Cov(\hat{\beta}_i, \hat{\beta}_j) = \zeta_{ij}\sigma^2$ 이며, 추정량들의 벡터 $\overrightarrow{\hat{\beta}}$는 다변수(multivariate) 정규분포를 갖는다.
마지막으로, $\hat{\sigma}^2$는 $\overrightarrow{\hat{\beta}}$에 독립이고, $n\hat{\sigma}^2/\sigma^2$는 $n-p$ 자유도인 χ^2 분포를 따른다.

§. 일반선형모형에서의 가설검정 (Testing Hypotheses)

이 장에서의 가설검정의 내용은 단순선형회귀와 거의 같다. 검정통계량과 좀 더 일반화된 가설을 제외하면 전반적인 흐름이 유사하므로 이 부분은 교재를 참고하는 것을 권한다. (page 745-747)

§. 일반선형모형에서의 예측 (Prediction)

이 장도 마찬가지로 단순선형회귀에서 다루었던 내용과 마찬가지의 흐름으로 진행된다. 단, 정의되는 식들이 모두 행렬과 벡터를 이용하여 표현되는 것이 다르다. 결과적으로 논의는 매우 유사하므로, 교재를 참고하도록 한다. (page 747-748)

§. 결정계수 R^2 (Multiple R^2)

다중선형회귀의 문제에서, 우리는 일반적으로 변수 $X_1, ..., X_n$이 확률변수 Y를 얼마나 잘 설명할 수 있는가를 결정하기 위한 것에 관심이 있다. 따라서 우리는 이러한 것을 결정하기 위해 **'결정계수'**의 개념을 도입한다. 즉, 추정한 선형 모형이 주어진 자료에 적합한 정도를 재는 척도로써 반응변수 Y의 변동량 중에서 적용한 모형으로 설명가능한 부분의 비율을 가리킨다.

일반적으로, **모형의 설명력**으로 해석되지만 모형에 설명변수가 들어갈수록 증가하기 때문에 해석에 주의해야 한다. 이러한 문제를 해결하기 위해서 **조정 결정계수(adjusted R^2)**가 제시되었다.

Definition **결정계수(coefficient of determination)**

결정계수는 "전체 제곱합 중에서 회귀 제곱합이 설명하는 비중"을 일컫는다. 즉, $SST = \Sigma (y_i - \bar{y})^2$, $SSR = \Sigma (y_i - \hat{y})^2$로 두면

$$R^2 = \frac{SSR}{SST} \ , \ 0 \leq R^2 \leq 1$$

이는 0에 가까울수록 모형의 설명력이 떨어지고 1에 가까울수록 모형의 설명력이 높음을 의미한다. 단순선형회귀에서는 결정계수를 사용함에 무리가 없으나 다중선형회귀의 경우는 종속변수의 변동을 별로 설명하지 못하는 변수가 추가되더라도 결정계수값이 커지는 문제가 발생하므로 **조정 결정계수**를 사용한다. 그 식은 다음과 같다.

$$R^2 = 1 - (n-1)\frac{MSE}{SST} \ , \ MSE \text{ (Mean squared error)는 잔차제곱합을 일컫는다.}$$

▷ **예제 11.5.8 : Unemployment in the 1950s.**

결정계수의 정의를 이용하여 이 예제에서의 회귀 모형의 결정계수를 구해본다. 교재의 방법에 따르면 결정계수의 값은 0.8656으로 구해진다. R 프로그래밍을 통하여 구해보면 다음과 같다.

Results

```
>df_unemployment=data.frame(Unemploy,product,year)
> df_unemployment
   Unemploy product year
1       3.1     113    1
2       1.9     123    2
3       1.7     127    3
4       1.6     138    4
5       3.2     130    5
6       2.7     146    6
7       2.6     151    7
8       2.9     152    8
9       4.7     141    9
10      3.8     159   10

> # Multiple Linear Model
> model=lm(y ~ x1 + x2)
> model

Call:
lm(formula = y ~ x1 + x2)

Coefficients:
(Intercept)           x1           x2
    13.4539      -0.1033       0.6594

> # R_squared (the coefficient of determination)
> summary(model)$r.squared
[1] 0.8655401
```

R Code

```
## Table
Unemploy<-c(3.1,1.9,1.7,1.6,3.2,2.7,2.6,2.9,4.7,3.8)
product<-c(113,123,127,138,130,146,151,152,141,159)
year<-c(1,2,3,4,5,6,7,8,9,10)

df_unemployment=data.frame(Unemploy,product,year)
df_unemployment

# Variables
x1=df_unemployment$product
x2=df_unemployment$year
y=df_unemployment$Unemploy

# Multiple Linear Model
model=lm(y ~ x1 + x2)
model

# R_squared (the coefficient of determination)
summary(model)$r.squared
```

▲

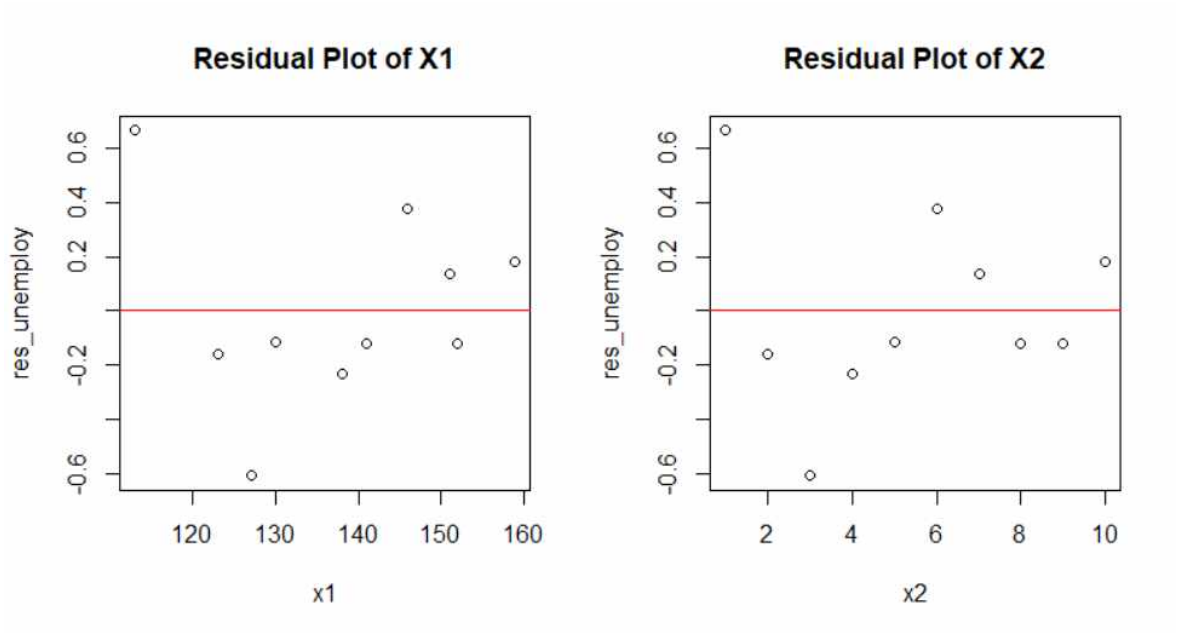
§. 일반선형모형에서의 잔차 분석 (Analysis of Residuals)

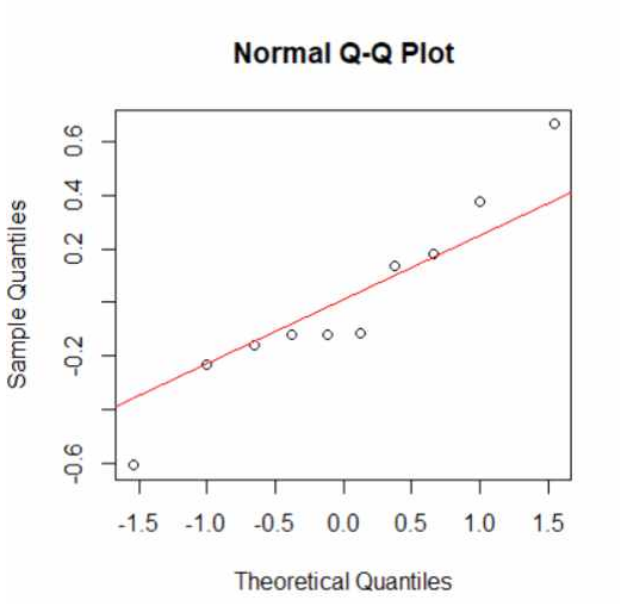
▷ **예제 11.5.9 : Unemployment in the 1950s.**

단순선형회귀에서의 잔차 분석과 마찬가지로, 잔차에 대한 일반화를 적용할 수 있다. 즉,

$$e_i = y_i - \hat{y}_i = y_i - z_{i0}\beta_0 - \cdots - z_{ip-1}\beta_{p-1}$$

이제 예제에서의 정보를 이용하자. $p = 3$, $z_{i0} = 1$ for $\forall\, i$ 이고, Table의 정보를 이용하여 회귀모형을 구한다. R 프로그래밍을 통하여 잔차 그래프를 그려본다. 교재에서는 년도의 첫 번째 값에 해당하는 행을 제외한 플롯도 제시하였으나 생략하도록 한다. Q-Q 플롯도 마찬가지로 나머지 결과들은 교재를 참고하도록 한다.





▲

11.6 분산분석(Analysis of Variance)

§. 일원배치분산분석(The One-way Layout / One-way ANOVA)

Definition of One-way ANOVA

회귀분석의 한 형태로써, Analysis of Variance의 약자이다. 각 관찰값과 평균 간 차이인 편차를 제곱해 합산한 후 표본크기로 나눈 분산을 이용해 2개 이상 집단 간 평균 차이를 검증하는 방법이다. 이 분석은 독립변수가 2개 이상 범주 수준으로 측정된 질적 데이터이고, 종속변수가 유사등간수준 이상으로 측정된 양적 데이터일 경우에 사용할 수 있다. 분산분석은 k개 집단들의 평균이 모집단 평균과 동일하다는 귀무가설을 검증한다. 즉, 다음과 같은 가설을 검증하고자 한다.

$$H_0 : \mu_1 = \cdots = \mu_k$$
$$H_1 : H_0 \text{ is not true}$$

이 때, 각 μ_i는 각 모집단의 평균을 나타낸다. 분산분석은 회귀분석의 특별한 형태이므로, 회귀분서고가 분산분석의 작동원리는 거의 동일하다. 여기서 일원분산분석은 **관측값에 대한 한 종류의 인자만의 영향을 조사**하고자 할 때 사용하는 방법이다.

분산분석을 위한 가정으로는 **표본의 독립성, 반응변수의 정규성, 반응변수의 등분산**을 가정한다.

▷ **예제 11.6.1-2 : Calories in Hot Dogs.**

Table 11.15의 데이터는 63개의 핫도그 브랜드의 칼로리에 대한 내용을 이루고 있다. 핫도그의 종류는 소고기(beef), 잡육(meat), 가금류(poultry, 길러진 닭 또는 조류), 특제(specialty, 칠리나 치즈로 채워진 소세지)로 구분된다.

Table 11.15 Calorie counts in four types of hot dogs for Example 11.6.2

Type	Calorie Count
Beef	186, 181, 176, 149, 184, 190, 158, 139, 175, 148, 152, 111, 141, 153, 190, 157, 131, 149, 135, 132
Meat	173, 191, 182, 190, 172, 147, 146, 139, 175, 136, 179, 153, 107, 195, 135, 140, 138
Poultry	129, 132, 102, 106, 94, 102, 87, 99, 107, 113, 135, 142, 86, 143, 152, 146, 144
Specialty	155, 170, 114, 191, 162, 146, 140, 187, 180

이 예제에서, 표본크기는 각각 $n_1 = 20$ (beef), $n_2 = 17$ (meat), $n_3 = 17$ (poultry), $n_4 = 9$ (specialty)이다. 이 경우에, 우리는 μ_1을 beef 핫도그의 평균 칼로리로 두자. 마찬가지로 μ_2, μ_3, μ_4도 각각의 Type의 평균 칼로리라고 하자. 모든 칼로리는 독립인 정규 확률변수로써 분산을 σ^2로 갖는다고 가정한다. 이러한 데이터는 ANOVA 방법론으로써 분석될 것이다.

▲

1. 일원배치법의 자료구조

	처리 1	처리 2	…	처리 I
반복 측정된 관측값	y_{11}	y_{21}	…	y_{I1}
	y_{12}	y_{22}	…	y_{I2}
	⋮	⋮	…	⋮
평균	y_{1n_1}	y_{2n_2}	⋮	y_{In_I}
	$\bar{y}_1$	$\bar{y}_2$	…	$\bar{y}_I$

n_i : i 번째 처리에서 관측값의 개수 ($N = \sum_{i=1}^{I} n_i$: 전체 관측값의 개수)

y_{ij} : i 번째 처리의 j 번째 관측값

$\bar{y}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} y_{ij}$, $i = 1,2,\cdots,I,$ $\bar{y} = \frac{1}{N}\sum_{i=1}^{I}\sum_{j=1}^{n_i} y_{ij}$: 전체 평균

2. 일원배치법의 모형 (반복수가 같은 경우)

$$Y_{ij} = \mu_i + \epsilon_{ij} = \alpha_i + \epsilon_{ij} \ , \ i = 1, \cdots, I, \ j = 1, \cdots, n_i$$

ϵ_{ij} : 실험의 오차, $\epsilon_{ij} \sim N(0, \sigma^2)$, 서로 독립, μ_i : i 번째 처리의 모평균, μ_0 : 총 평균, α_i : i 번째 처리에 의한 효과(주효과)

3. 관심 : 처리들의 효과가 동일한지가 궁금

가설 : $H_0 : \mu_1 = \cdots = \mu_I$ 또는 $H_0 : \alpha_1 = \cdots = \alpha_I$, $\hat{\mu}_i = \bar{y}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} y_{ij}.$

$\Rightarrow$ I 개의 추정치들을 동시에 비교하기 위해 $\sum_{i=1}^{I}(\bar{y}_i - \bar{y})^2$를 고려한다.

(제곱합이 큰 값을 가질 때 처리 평균값들은 전체 평균으로부터 멀리 흩어져 있어 서로 매우 다른 값을 가짐을 의미하고, 이에 반해 제곱합이 작은 값일때는 서로 유사한 값을 가짐을 의미함.)

4. 검정통계량을 유도 : 총제곱합의 분해(Partitioning)

$$y_{ij} = \bar{y} + (\bar{y}_i - \bar{y}) + (y_{ij} - \bar{y}_i) \ \Rightarrow \ \sum_{i=1}^{I}\sum_{j=1}^{n_i}(y_{ij} - \bar{y})^2 = \sum_{i=1}^{I}\sum_{j=1}^{n_i}(\bar{y}_i - \bar{y})^2 + \sum_{i=1}^{I}\sum_{j=1}^{n_i}(y_{ij} - \bar{y}_i)^2$$

$$\Rightarrow \ SST = SSTR + SSE$$

SST : **관측값의 총제곱합**, $SSTR$: **처리(treatment)제곱합**, SSE : **오차제곱합**

$$\therefore \ \textbf{검정통계량} : \ F^* = \frac{SSTR/(I-1)}{SSE/(N-I)} = \frac{MSTR}{MSE} \sim F(I-1, N-1)$$

[일원배치법의 분산분석표]

요인	제곱합	자유도	평균제곱	F값	유의확률
처리	$SSTR$	$I-1$	$MSTR=\frac{SSTR}{I-1}$	$F^*=\frac{MSTR}{MSE}$	$P(F \geq F^*)$
잔차	SSE	$N-I$	$MSE=\frac{SSE}{N-I}$		
계	SST	$N-1$			

[제곱합의 간편 계산식]

$T_i=\sum_{j=1}^{n_i} y_{ij}$: 처리 i 에서의 모든 관측값의 합계

$T=\sum_{i=1}^{I} T_i$: 모든 관측값의 총계

$$SST=\sum_{i=1}^{I}\sum_{j=1}^{n_i} y_{ij}^2-\frac{T^2}{n}\ ,\ n=\sum_{i=1}^{I} n_i$$

$$SSTR=\sum_{i=1}^{I}\frac{T_i^2}{n_i}-\frac{T^2}{n}$$

$$SSE=\sum_{i=1}^{I}\sum_{j=1}^{n_i} y_{ij}^2-\sum_{i=1}^{I}\frac{T^2}{n_i}=SST=SSTR$$

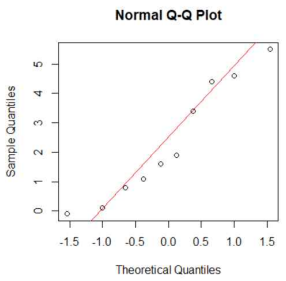
▷ **예제 11.6.6 : Calories in Hot Dogs.**

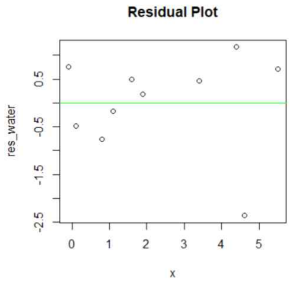
전반적인 R 프로그래밍 코드와 실행 결과로 대체하겠다. 그 결과는 다음 페이지에 제시하겠다. 단, 다음 페이지에서 구한 분산분석 결과로 가설검정에 대한 결과만 서술하면, ANOVA 표에서 검정통계량 F^*의 값이 11.6 이고, p-값은 4.48×10^{-6}로 구해지므로, 이는 매우 작은 값이고, 유의수준 0.001 보다도 작으므로(0.00000448) 유의수준 0.001에서 귀무가설을 기각할 수 있다. 따라서 각 핫도그 브랜드 별 칼로리 평균에 차이가 있다고 할 수 있다. 추가적으로 코드 페이지 맨 마지막에 등분산성을 검정하였는데 (Barlett test), 검정 결과 p-값인 0.9582는 유의수준 0.001보다 훨씬 큰 값이다. 이는 곧 등분산성을 잘 만족한다는 것을 의미한다.

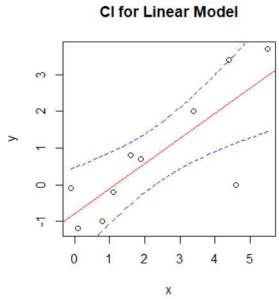
▲

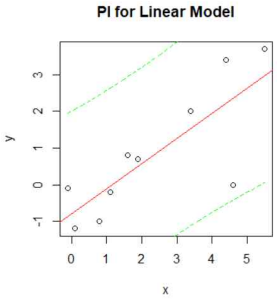
[Overall Codes for The Example 11.1 in R]

Results









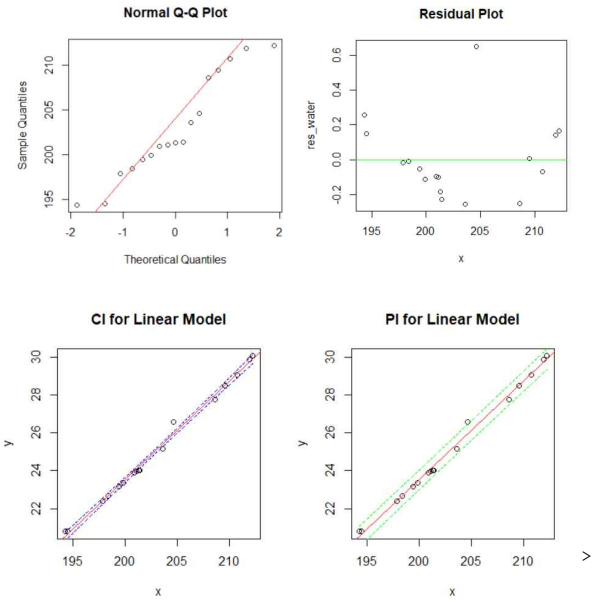

```
> # Prediction of specific value
> Linefunc(3.24)
[1] 1.433388

> # Simple Linear Model
> model
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
    -0.7861       0.6850

> # Significant Test for linear regression
> summary(model)

Call:
lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.3651 -0.4036  0.3208  0.6613  1.1720

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.7861     0.5418  -1.451  0.18485
x             0.6850     0.1802   3.801  0.00523 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.083 on 8 degrees of freedom
Multiple R-squared:  0.6436,    Adjusted R-squared:  0.599
F-statistic: 14.45 on 1 and 8 DF,  p-value: 0.005231
```

R Code

```
# Table
drug_A<-c(1.9,0.8,1.1,0.1,-0.1,4.4,4.6,1.6,5.5,3.4)
drug_B<-c(0.7,-1.0,-0.2,-1.2,-0.1,3.4,0.0,0.8,3.7,2.0)
df_blood=data.frame(drug_A,drug_B)
df_blood

# Variables
x=df_blood$drug_A
y=df_blood$drug_B

# Scatter Plot
scatter_blood=plot(x,y)

# Q-Q Plot
qqnorm(x)
qqline(x,distribution=qnorm,col="red")

# Simple Linear Model
model=lm(y ~ x)
plot(x,y,main="Linear Regression")
abline(model,col="blue")

# Residual Plot
res_water=resid(model)
plot(x,res_water,main="Residual Plot")
abline(0,0,col="green")

#Linear function
Linefunc<-function(x){
-0.7861478+0.6850420*x
}

# Prediction of specific value
Linefunc(3.24)

par(mfrow=c(1,2))
# 95% Confidence Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),interval="confidenc
e",level=0.95)
plot(x,y,main="CI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="blue",lty=2)
lines(newx,CI[,3],col="blue",lty=2)

# 95% Prediction Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),interval="prediction
",level=0.95)
plot(x,y,main="PI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="green",lty=2)
lines(newx,CI[,3],col="green",lty=2)

# Significant Test for linear regression
summary(model)
```

[Overall Codes for The Example 11.2-3 in R]

Results



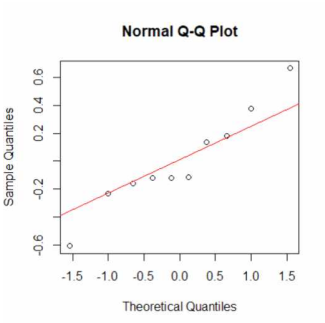

```
> # Prediction of specific value
> Linefunc(201.5)
[1] 24.29909

> # Simple Linear Model
> model=lm(y ~ x)
> model

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
   -81.0637       0.5229

> # Significant Test for linear regression
> summary(model)

Call:
lm(formula = y ~ x)

Residuals:
      Min        1Q    Median        3Q       Max
-0.25717 -0.11246 -0.05102  0.14283  0.64994

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -81.06373    2.05182  -39.51   <2e-16 ***
x             0.52289    0.01011   51.74   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2328 on 15 degrees of freedom
Multiple R-squared:  0.9944,    Adjusted R-squared:  0.9941
F-statistic:  2677 on 1 and 15 DF,  p-value: < 2.2e-16
```

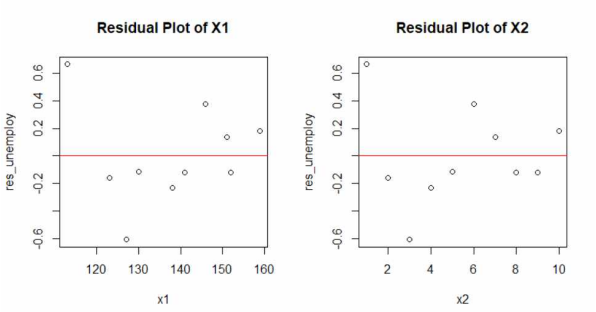
R Code

```
# Table
Boil<-c(194.5,194.3,197.9,198.4,199.4,199.9,200.9,201.1,201.4,201.3,
203.6,204.6,209.5,208.6,210.7,211.9,212.2)
Pres<-c(20.79,20.79,22.40,22.67,23.15,23.35,23.89,23.99,24.02,24.01,
25.14,26.57,28.49,27.76,29.04,29.88,30.06)
df_water=data.frame(Boil,Pres)
df_water

# Variables
x=df_water$Boil
y=df_water$Pres

# Scatter Plot
scatter_water=plot(x,y)

# Q-Q Plot
qqnorm(x)
qqline(x,distribution=qnorm,col="red")

# Simple Linear Model
model=lm(y ~ x)
plot(x,y,main="Linear Regression")
abline(model,col="blue")

# Residual Plot
res_water=resid(model)
plot(x,res_water,main="Residual Plot")
abline(0,0,col="green")

#Linear function
Linefunc<-function(x){
-81.0637271+0.5228924*x
}

# Prediction of specific value
Linefunc(201.5)

par(mfrow=c(1,2))
# 95% Confidence Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),interval="confidenc
e",level=0.95)
plot(x,y,main="CI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="blue",lty=2)
lines(newx,CI[,3],col="blue",lty=2)

# 95% Prediction Inverval
newx=seq(min(x),max(x),by=0.05)
CI=predict(model,newdata=data.frame(x=newx),interval="prediction
",level=0.95)
plot(x,y,main="PI for Linear Model")
abline(model,col="red")
lines(newx,CI[,2],col="green",lty=2)
lines(newx,CI[,3],col="green",lty=2)

# Significant Test for linear regression
summary(model)
```

[Overall Codes for The Example 11.5 in R]

Results





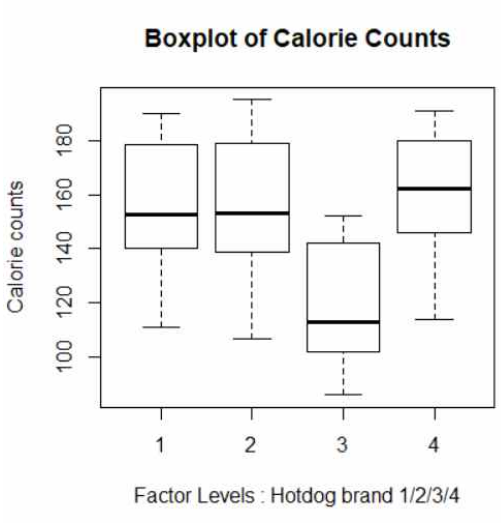

```
> df_unemployment=data.frame(Unemploy,product,year)
> df_unemployment
   Unemploy product year
1       3.1     113    1
2       1.9     123    2
3       1.7     127    3
4       1.6     138    4
5       3.2     130    5
6       2.7     146    6
7       2.6     151    7
8       2.9     152    8
9       4.7     141    9
10      3.8     159   10

> # Multiple Linear Model
> model=lm(y ~ x1 + x2)
> model

Call:
lm(formula = y ~ x1 + x2)

Coefficients:
(Intercept)           x1           x2
    13.4539      -0.1033       0.6594

> # R_squared (the coefficient of determination)
> summary(model)$r.squared
[1] 0.8655401
```

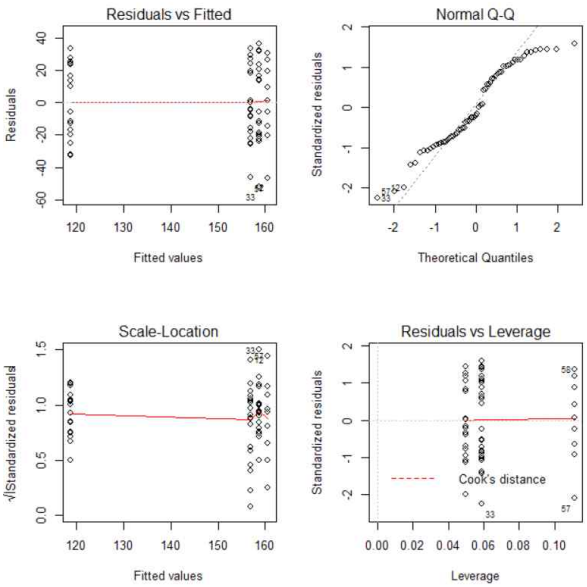
R Code

```
# Table
Unemploy<-c(3.1,1.9,1.7,1.6,3.2,2.7,2.6,2.9,4.7,3.8)
product<-c(113,123,127,138,130,146,151,152,141,159)
year<-c(1,2,3,4,5,6,7,8,9,10)

df_unemployment=data.frame(Unemploy,product,year)
df_unemployment

# Variables
x1=df_unemployment$product
x2=df_unemployment$year
y=df_unemployment$Unemploy

# Multiple Linear Model
model=lm(y ~ x1 + x2)
model

# R_squared (the coefficient of determination)
summary(model)$r.squared

# Q-Q Plot
qqnorm(res_unemploy)
qqline(res_unemploy,distribution=qnorm,col="red")

par(mfrow=c(1,2))
# Residual Plot_x1
res_unemploy=resid(model)
plot(x1,res_unemploy,main="Residual Plot of X1")
abline(0,0,col="red")

# Residual Plot_x2
res_unemploy=resid(model)
plot(x2,res_unemploy,main="Residual Plot of X2")
abline(0,0,col="red")
```

[Overall Codes for The Example 11.6 in R]

Results






```
> summary(analysis_dog)
             Df Sum Sq Mean Sq F value    Pr(>F)
group         3  19454    6485    11.6 4.48e-06 ***
Residuals    59  32995     559
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.

> # Bartlett test to test the null hypothesis of equal group variances
> bartlett.test(y ~ group, data = Hotdog_df)

        Bartlett test of homogeneity of variances

data:  y by group
Bartlett's K-squared = 0.30971, df = 3, p-value = 0.9582
```

R Code

```
##---------------------------------------------------------
## One-way ANOVA : aov(), oneway.test
##---------------------------------------------------------

## Are there any daily outcome differences among temperature conditions?
# group 1 : Beef
# group 2 : Meat
# group 3 : Poultry
# group 4 : Specialty

# daily outcome by calorie counts (group 1/2/3/4)
y1<-c(186, 181, 176, 149, 184, 190, 158, 139, 175, 148, 152, 111, 141, 153, 190, 157, 131, 149, 135, 132)
y2<-c(173, 191, 182, 190, 172, 147, 146, 139, 175, 136, 179, 153, 107, 195, 135, 140, 138)
y3<-c(129, 132, 102, 106, 94, 102, 87, 99, 107, 113, 135, 142, 86, 143, 152, 146, 144)
y4<-c(155, 170, 114, 191, 162, 146, 140, 187, 180)

y<-c(y1,y2,y3,y4)
n<-c(20,17,17,9)
group<-rep(1:4,n)
group

# combining into data.frame
Hotdog_df <- data.frame(y, group)
Hotdog_df

sapply(Hotdog_df, class)

# transform from 'integer' to 'factor'
Hotdog_df <- transform(Hotdog_df, group = factor(group))
sapply(Hotdog_df, class)

# boxplot
attach(Hotdog_df)
boxplot(y ~ group, main = "Boxplot of Calorie Counts",
                xlab = "Factor Levels : Hotdog brand 1/2/3/4",
                ylab = "Calorie counts")

# descriptive statistics by group
tapply(y, group, summary)

detach(Hotdog_df)

# One-Way ANOVA
analysis_dog = aov(y ~ group, data=Hotdog_df)
summary(analysis_dog)

# Bartlett test to test the null hypothesis of equal group variances
bartlett.test(y ~ group, data = Hotdog_df)

# Plots
par(mfrow=c(2,2))
plot(analysis_dog)
```

[Case study in R]

주어진 데이터에 대한 통계 분석을 실시해본다. 우선, 주어진 데이터 table_7_3.csv를 통계 소프트웨어인 R로 불러오고자 한다. 데이터를 불러오기 전에, 우선 데이터가 있는 경로를 설정해주어야 한다. 그 방법은 다음과 같다.

Results

```
> # 현재 경로 확인
> getwd()
[1] "C:/Users/SangmanJeong/Documents“

> # 새로운 경로 지정
> setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")

> # 현재 경로 재확인
> getwd()
[1] "C:/Users/SangmanJeong/Desktop/18년 2학기/통계학
```

R Code

```
# 현재 경로 확인
getwd()

# 새로운 경로 지정
setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")

# 현재 경로 재확인
getwd()
```

이 때, 경로 지정 시 \는 /로 바꾸어주고, 경로 전체를 “”로 묶어주어야 함을 주의하자. 이제 설정한 경로에 불러올 데이터를 넣고 이를 불러온다. 그 방법은 아래와 같다.

Results

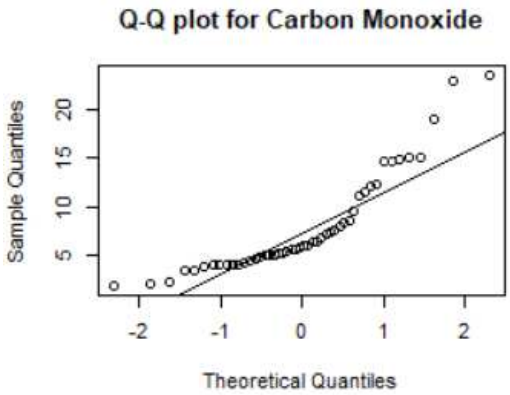
```
> engine <- read.csv(file="table_7_3.csv",sep =",",head = TRUE)
> names(engine)
[1] "en"  "hc"  "co"  "nox"
> summary(engine)
       en                hc                co               nox
 Min.   : 1.00   Min.   :0.3400   Min.   : 1.850   Min.   :0.490
 1st Qu.:12.75   1st Qu.:0.4375   1st Qu.: 4.388   1st Qu.:1.110
 Median :24.50   Median :0.5100   Median : 5.905   Median :1.315
 Mean   :24.00   Mean   :0.5502   Mean   : 7.879   Mean   :1.340
 3rd Qu.:35.25   3rd Qu.:0.6025   3rd Qu.:10.015   3rd Qu.:1.495
 Max.   :46.00   Max.   :1.1000   Max.   :23.530   Max.   :2.940
```

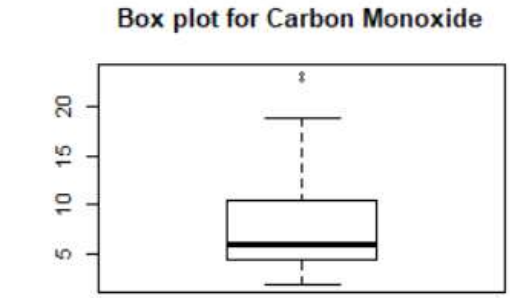
R Code

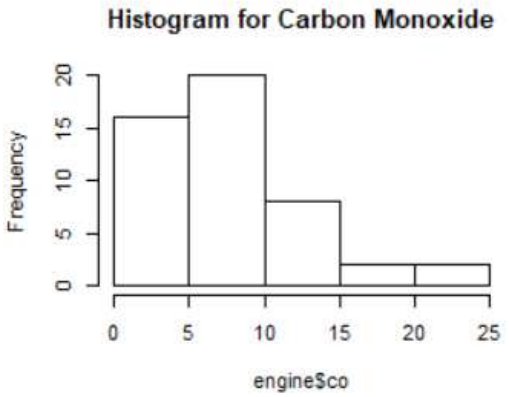
```
# csv 데이터 불러오기
engine <- read.csv(file="table_7_3.csv",sep =",",head = TRUE)

# 불러온 데이터의 컬럼명 보기
names(engine)

# 데이터의 전반적인 기술 통계량 요약
summary(engine)
```

데이터를 불러온 후 간단한 기술 통계량을 확인하였다. csv 데이터가 아닌 엑셀 데이터를 불러오고자 한다면 위의 명령어 read.csv에서 read.xlsx 로 입력하면 된다. 이제 주어진 데이터를 시각적으로 분석해보고자 한다. 다음의 명령어들을 입력한다. 특정 결과 중 코드 란에 여백이 많은 경우 여백 활용을 위해 관련 분석은 코드 란에 파란 색으로 기술할 것이다.

Results







R Code

```
# Q-Q 플롯
qqnorm(engine$co,main="Q-Q plot for Carbon Monoxide")
qqline(engine$co)

# Box 플롯
boxplot(engine$co,main="Box plot for Carbon Monoxide")

# 히스토그램
hist(engine$co,main="Histogram for Carbon Monoxide")

# 서브플롯
par(mfrow=c(3,1))
qqnorm(engine$co,main="Q-Q plot for Carbon Monoxide")
qqline(engine$co)
boxplot(engine$co,main="Box plot for Carbon Monoxide")
hist(engine$co,main="Histogram for Carbon Monoxide")
```

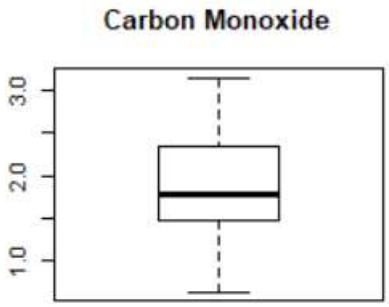
※ Q-Q 플롯은 주어진 데이터의 Carbon Monoxide 컬럼에 대한 정규성을 확인한 것이다. 전반적으로 정규성을 만족하는 직선에 가깝게 분포되어있으나 오른쪽으로 갈수록 점들이 멀어지는 경향을 보인다.

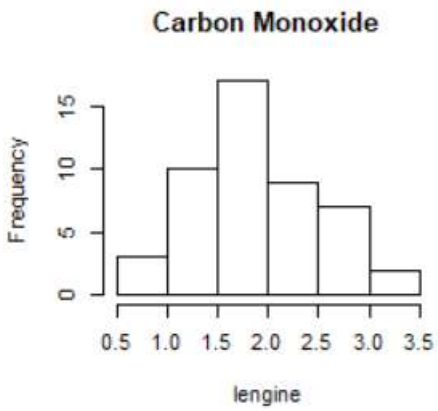
※ Box 플롯은 시각적으로 데이터의 분포를 보는 방법이다. 특히 사분위수와 이상점의 유무를 확인하는데 적합하다. 출력한 플롯을 보면 데이터의 사분위수 범위를 넘어선 이상점이 보이고, 중앙값에서 1사분위수와 3사분위수의 범위가 크게 차이가 남을 확인할 수 있다.

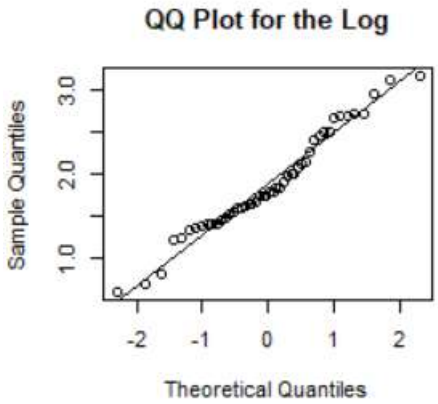
※ 히스토그램은 데이터의 분포가 어떤 모양을 띄는지 간략히 확인할 수 있는 그래프이다. 출력한 히스토그램을 보면, 정규성을 띈다고 보기 어려울 수 있다. 아주 큰 시각에서 본다면 어느정도 정규분포적인 느낌을 따른다고 볼 수 있으나, Q-Q 플롯과 마찬가지로 몇몇 이상점의 유무에 따라 달라질 수 있다.

앞서 데이터를 시각화하여 분석해보았다. 그런데 주어진 데이터가 정규성을 따르지 않는 것처럼 보인다. 우리는 데이터가 정말 정규성을 띄지 않는지 확인해보고 싶으므로, 데이터를 로그 스케일로 변환하여 다시 그래프를 그려보도록 한다. 로그 스케일을 사용하는 이유는 데이터의 수치적 값에 의존하지 않고 그 본질적인 비율을 보고자 함이다. 실제로 로그 스케일로 출력했을 때 데이터가 정규성을 따르는지는 아래에 출력한 그래프로써 확인할 수 있다.

Results







R Code

```
# Q-Q 플롯
qqnorm(engine$co,main="Q-Q plot for Carbon Monoxide")
qqline(engine$co)

# Box 플롯
boxplot(engine$co,main="Box plot for Carbon Monoxide")

# 히스토그램
hist(engine$co,main="Histogram for Carbon Monoxide")

# 로그 스케일 플롯
par(mfrow=c(3,1))
lengine <- log(engine$co)
boxplot(lengine,main="Carbon Monoxide")
hist(lengine,main="Carbon Monoxide")
qqnorm(lengine,main="QQ Plot for the Log")
qqline(lengine)
```

※ 로그 스케일로 그래프를 출력하였다. 그래프를 살펴보면, 박스 플롯의 이상점이 제외되었고, 분위수의 범위도 조정된 것을 확인할 수 있다. 또한 히스토그램은 이전의 그래프보다 훨씬 정규성을 띄는 분포를 보여주며 Q-Q 플롯 또한 강력하게 정규성을 띄는 것을 확인할 수 있다.

이제, 주어진 데이터의 신뢰구간을 구해보자. 우리는 기존의 컬럼값이 아닌 로그 스케일이 적용된 컬럼값으로 사용할 것이다. 신뢰구간을 구하기 위해 기본적인 데이터의 평균, 표준편차, 표본크기, 표준오차를 구해본다.

Results

```
 > m
[1] 1.883678
> s
[1] 0.5983851
> n
[1] 48
> se
[1] 0.08636945
```

R Code

```
# 평균, 표준편차, 표본크기
m <- mean(lengine)
s <- sd(lengine)
n <- length(lengine)

# 표준오차
se <- s/sqrt(n)
```

우리는 95% 신뢰수준에서 신뢰구간을 구할 것이다. 우선 신뢰구간의 주변오차(margin of error)를 구해보자. 신뢰구간에 대한 내용은 $X_1, ..., X_n \sim N(\mu,\sigma^2)$, μ, σ^2 : unknown, $\theta=(\mu,\sigma^2)$, $g(\theta)=\mu$ 을 가정했을 때, $\overline{X}_n=\frac{1}{n}\sum_{i=1}^{n}X_i$, $\sigma'=\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\overline{X}_n)^2}$ 인 통계량(사실, M.L.E.인)을 이용하여 다음과 같이 $g(\theta)$에 대한 계수 γ의 신뢰구간을 구할 수 있다. 여기서 수준은 $\alpha_0=1-\gamma$ 이다.

$$I=\left(\overline{X}_n-T_{n-1}^{-1}\left(\frac{1+\gamma}{2}\right)\frac{\sigma'}{\sqrt{n}},\ \overline{X}_n+T_{n-1}^{-1}\left(\frac{1+\gamma}{2}\right)\frac{\sigma'}{\sqrt{n}}\right)\ ,\ \text{where}$$

$T_{n-1}^{-1}(\bullet)$: the quantile function of the t-distribution with $n-1$ degrees of freedom.

이를 이제 R 프로그래밍을 통하여 구해보자.

Results

```
> # 신뢰구간에서의 주변오차
> # qt()는 t-분포의 분계(quantile)함수
> error <- se*qt(0.975,df=n-1)
> error
[1] 0.1737529

> # 구간의 양끝 주변오차
> left <- m - error
> right <- m + error

> # 95% 신뢰구간
> CI_95<-c(left, right)
> CI_95
[1] 1.709925 2.057431
```

R Code

```
# Note : t-분포의 밀도, 확률, 분계, 무작위추출 함수
## dt(x, df, ncp, log = FALSE)
## pt(q, df, ncp, lower.tail = TRUE, log.p = FALSE)
## qt(p, df, ncp, lower.tail = TRUE, log.p = FALSE)
## rt(n, df, ncp)

# 신뢰구간에서의 주변오차
# qt()는 t-분포의 분계(quantile)함수
error <- se*qt(0.975,df=n-1)
error

# 구간의 양끝 주변오차
left <- m - error
right <- m + error

# 95% 신뢰구간
CI_95<-c(left, right)
CI_95
```

프로그래밍을 통하여 구한 결과, 신뢰구간은 $(1.709925, 2.057431)$로 구해진다. 그런데 이는 로그 스케일로 변환된 데이터 값에 대한 신뢰구간이므로, 신뢰구간의 값을 다시 원래 값으로 변환해주기 위해 exponential을 취하면 다음의 원래의 신뢰구간 값인 $(5.528548, 7.825840)$을 얻을 수 있다.

Results

```
> # 95% 신뢰구간_로그 스케일 해제
> CI_95<-c(exp(left), exp(right))
> CI_95
[1] 5.528548 7.825840
```

R Code

```
# 95% 신뢰구간_로그 스케일 해제
CI_95<-c(exp(left), exp(right))
CI_95
```

이제 주어진 데이터로 가설 검정을 실시하고자 한다. 모평균에 대한 가설 검정을 실시한다. 이 때, 가설은 양측 검정(Two-sided)으로 검정한다. 우리는 가설을 다음과 같이 세우고 이를 검정하고자 한다.

$$H_0 : \mu_x = 5.4 \text{ vs } H_1 : \mu_x \neq 5.4$$

가설검정을 실행하기 위해서는, 우리가 귀무가설이 참이라는 가정 하에 가정된 평균(5.4)의 신뢰구간을 찾아야만 한다. 그런데 우린 이미 이에 대한 모평균의 신뢰구간을 찾았었다. 이를 이용하면 여기서 평균을 5.4로 가정한 이유를 설명할 수 있다. 이전의 로그 스케일로 계산된 모평균의 신뢰구간은 $(1.709925, 2.057431)$ 이었고, 우리가 가정한 표본평균 5.4에 대한 로그 스케일의 신뢰구간은 $(1.512646, 1.860152)$으로 표본평균은 모평균의 신뢰구간에 포함되지 않는다. 따라서 우리는 귀무가설을 기각할 수 있다. 이는 실제로 모평균이 5.4인 경우 우리가 가정한 표본평균 값을 얻을 확률이 낮다는 의미와 같다. 이에 대한 간단한 프로그래밍 코드는 아래와 같다.

Results

```
> # 귀무가설의 신뢰구간
> CI_Null<-c(lNull , rNull)
> CI_Null
[1] 1.512646 1.860152
```

R Code

```
# 귀무가설에 대한 신뢰구간의 양 끝값
lNull <- log(5.4) - error
rNull <- log(5.4) + error

# 귀무가설의 신뢰구간
CI_Null<-c(lNull , rNull)
CI_Null
```

좀 더 자세한 방법으로 가설검정을 실시해보자. 신뢰구간을 이용하여 검정한 방법이 아닌 p-값을 이용하여 검정하는 방법이다. 이는 검정통계량을 구하여 유의수준에 따른 p-값과 비교하여 귀무가설의 기각 여부를 정하는 방법이다. 우선 p-값을 구해보자. 이는 다음과 같은 코드로 구할 수 있다.

Results

```
> # p-value
> 2*(1-pt((m-log(5.4))/se,df=n-1))

[1] 0.02692539
```

R Code

```
# p-value
# pt()는 t-분포의 분포함수, df : 자유도
# 양측검정이므로 2를 곱해준다.
2*(1-pt((m-log(5.4))/se,df=n-1))
```

p-값은 0.02692539, 약 2.7%로 구해졌다. 이는 우리가 정한 95% 신뢰수준, 즉 5%의 유의수준보다 작으므로 우린 귀무가설을 기각할 수 있다. 또 다른 방법으로, 직접 t-분포 검정을 이용하여 구하는 방법이 있다. 이는 앞선 논의들의 더욱 자세한 분석 방법이고, 이론적 배경은 완전히 동일하다. 다음의 코드를 입력하여 구할 수 있다. 자세한 설명은 코드 란에 기술하였다.

Results

```
> # t-test for the hypotheses
> t.test(lengine,mu = log(5.4),alternative = "two.sided")

        One Sample t-test

data:  lengine
t = 2.2841, df = 47, p-value = 0.02693
alternative hypothesis: true mean is not equal to 1.686399
95 percent confidence interval:
 1.709925 2.057431
sample estimates:
mean of x
 1.883678
```

R Code

```
# t-test for the hypotheses
t.test(lengine,mu = log(5.4),alternative = "two.sided")
```

lengine.mu 의 의미는 로그 스케일된 co 컬럼 값의 평균을 log(5.4)로 두겠다는 의미이고, 대립가설(alternative)를 양측검정(two-sided)로 하겠다는 의미이다.

코드 실행결과를 보면, p-값은 물론이고 검정통계량 t 값과 95% 신뢰구간, 대립가설의 결과 등을 더욱 자세히 제시함을 볼 수 있다.

또한 이는 단일 표본에 대한 모평균 가설 검정이므로 One Sample t-test 임을 알려주고 있다.

이제 검정력(power)을 이용하여 검정해보자. 우선, p-값(유의확률)은 귀무가설을 기각하게 하는 최소의 유의수준을 말한다(표본이 대립가설 방향으로 검정통계량의 값보다 더 어긋나게 될 확률이기도 하다). 이 때, 유의수준 α_0는 제 1종 오류의 최댓값으로써, "귀무가설이 참인데도 불구하고 이를 기각하는 확률"의 최댓값을 일컫는다.

검정력은 이와 반대로, 제 2종 오류를 이용한다. 즉 제 2종 오류를 β라 하면 $1-\beta$의 값이 검정력이다. 이는 대립가설이 사실일 때 귀무가설을 기각할 확률을 말한다. 이를 수식으로 설명하면 다음과 같다.

$$\beta = P(\text{제2종 오류}) = P(H_0 \text{ 채택} \mid H_1 \text{사실}),\ \text{검정력} : 1-\beta = P(H_0 \text{ 기각} \mid H_1 \text{사실})$$

검정력은 p-값, 즉 유의확률을 구하는 목적과 같다. 그러나 검정 과정에서 바라보는 시각을 제 2종 오류의 시각으로 구하고자 하는 것이다. 검정력 또한 앞서 했던 방법들처럼, 여러 방법으로 구할 수 있다.

검정력을 구하기 위해 우리는 평균을 설정하고 평균이 실제로 구한 값일 귀무가설을 받아들일 확률을 찾으면 된다. 첫 번째 소개할 방법은 비중심(non-central) t 검정을 사용할 수 없을 때 하는 방법이고 두 번째 소개할 방법은 비중심 t 검정을 사용할 수 있을 때의 방법을 소개한다. 마지막으로 세 번째 방법은 기존에 내장된 R의 기능을 이용하는 것이다.

우리는 우선, 만약 정말로 평균이 7이라면 귀무가설을 받아들일 확률을 구해야 한다. 실제 평균이 7이라고 가정하고 표본평균을 구하여 귀무가설이 참일 때 표본평균이 신뢰구간 내에 속할 확률을 구해보자. 우리는 계속하여 로그 스케일 값을 이용했으므로 7의 로그 스케일 값을 이용해야 함을 주의하고, 양측검정에 대한 내용을 다루고 있음을 유의하자. 첫 번째 방법의 코드는 다음과 같다.

Results

```
> # 신뢰구간의 왼쪽, 오른쪽 검정통계량 값
> tLeft <- (lNull-log(7))/(s/sqrt(n))
> tRight <- (rNull-log(7))/(s/sqrt(n))

> # 검정력
> p <- pt(tRight,df=n-1) - pt(tLeft,df=n-1)
> p
[1] 0.1629119
> 1-p
[1] 0.8370881
```

R Code

```
# 신뢰구간의 왼쪽, 오른쪽 검정통계량 T 값
tLeft <- (lNull-log(7))/(s/sqrt(n))
tRight <- (rNull-log(7))/(s/sqrt(n))

# 검정력
p <- pt(tRight,df=n-1) - pt(tLeft,df=n-1)
p
1-p
```

제 2종 오류의 영역의 확률은 약 16.3%으로 구해지고, 검정력은 83.7% 정도로 구해진다. 이는 대립가설이 참일 때 실제로 귀무가설을 기각할 확률, 즉 평균이 7임을 기각할 확률이 83.7% 임을 의미한다.

또 다른 방법으로, 앞서 서술했던 비중심 t-분포를 이용하는 방법을 소개한다. 이는 중심 검정분포와 달리 귀무가설이 참이라고 가정하지 않는다. 비중심 분포는 비중심 모수라 불리우는 여분의 모수를 갖게 되는데, 이 모수는 검정분포의 검정통계량의 변화율을 일컫는다. 이런 모수는 귀무가설에서부터 떨어져 있는 정도를 보여준다. 보편적으로 검정력 개념이 귀무가설이 거짓이라는 가정을 하기 때문에 비중심 분포를 이용하여 검정력을 구하는 것을 더 선호한다. 결과적으로, 요약하면 중심분포는 단일 모수와 자유도에 의해서 기술되지만 비중심 분포는 자유도와 검정통계량의 변화율(shifted)에 의해서 기술된다. 자세한 코드는 다음과 같이 하여 검정력을 구할 수 있다.

Results

```
> # t-분포의 확률분포함수
> pt(t,df=n-1,ncp=shift)-pt(-t,df=n-1,ncp=shift)
[1] 0.1628579
>
> # 검정력
> 1-(pt(t,df=n-1,ncp=shift)-pt(-t,df=n-1,ncp=shift))
[1] 0.8371421
```

R Code

```
# t-분포의 분계함수
t <- qt(0.975,df=n-1)
# 검정통계량의 변화율
shift <-(log(5.4)-log(7))/(s/sqrt(n))
# t-분포의 확률분포함수
pt(t,df=n-1,ncp=shift)-pt(-t,df=n-1,ncp=shift)
# 검정력
1-(pt(t,df=n-1,ncp=shift)-pt(-t,df=n-1,ncp=shift))
```

마지막으로, 검정력을 더욱 손쉽게, 그리고 자세하게 구할 수 있는 방법이 있다. 바로 R의 내장함수를 이용하는 것이다. 코드는 다음과 같다.

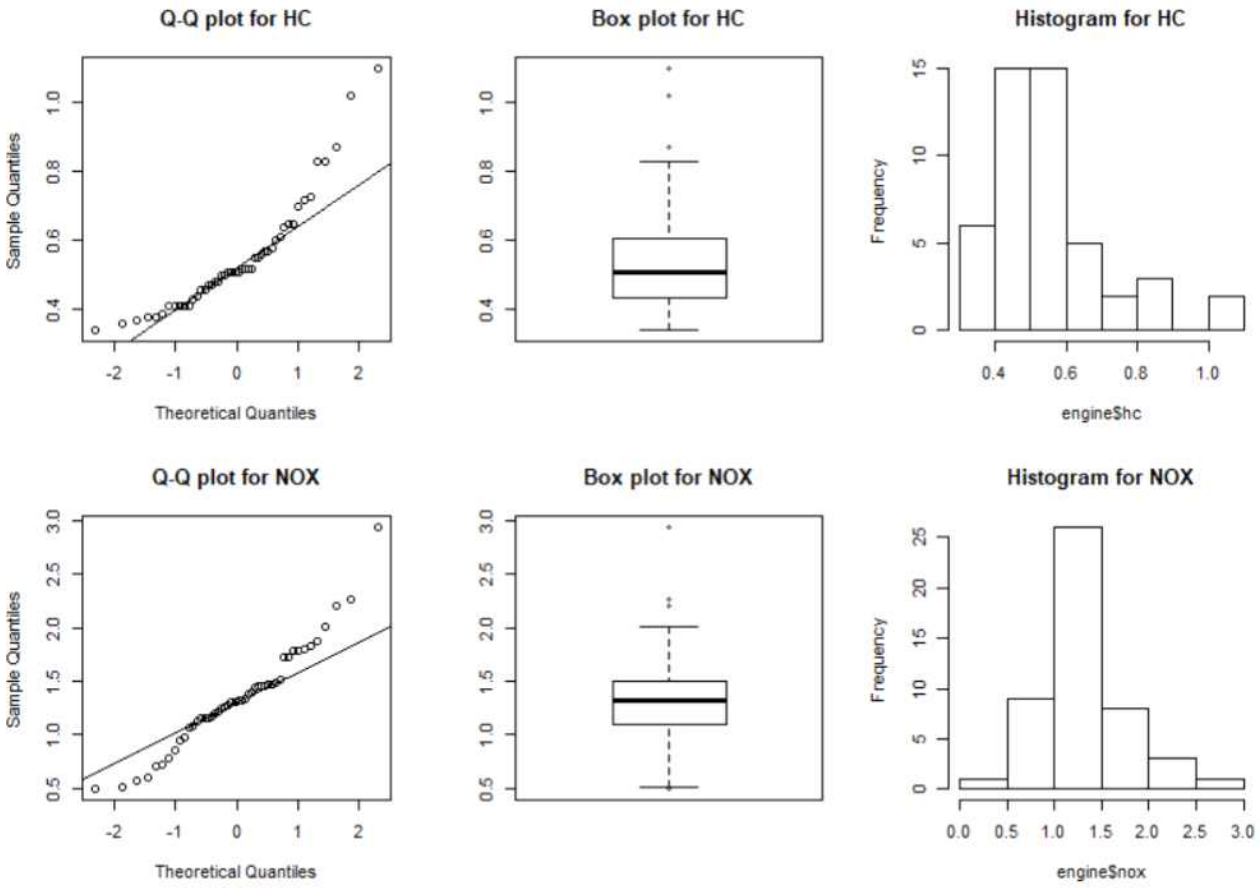
Results

```
> power.t.test(n=n,delta=log(7)-log(5.4),sd=s,sig.level=0.05,
+ type="one.sample",alternative="two.sided",strict = TRUE)

     One-sample t test power calculation

              n = 48
          delta = 0.2595112
             sd = 0.5983851
      sig.level = 0.05
          power = 0.8371421
    alternative = two.sided
```

R Code

```
# t-test of power
power.t.test(n=n,delta=log(7)-log(5.4),sd=s,sig.level=0.05,
type="one.sample",alternative="two.sided",strict = TRUE)
```

하나의 데이터를 가지고 다양한 통계적 분석을 해보았다. 본 Case Study에서는, engine의 co 컬럼에 대한 분석만 실시하였으나 다른 컬럼에 대해서도 마찬가지의 분석을 실행할 수 있다. 그러나 이는 이전의 분석과정과 마찬가지로, 나머지 컬럼에 대한 변수만 변경해주면 방법은 동일하므로, 이 과정을 다른 컬럼에 대해 다시 분석하는 것은 지엽적이다. 그러므로 우리는 다른 컬럼에 대한 시각화만 제시하고 이 장을 마치도록 한다. en 컬럼은 큰 의미가 없는 데이터이므로 제외하였다.



▲

[csdata.txt in R]

주어진 데이터에 대한 통계 분석을 실시해본다. 우선, 주어진 데이터 csdata.txt를 통계 소프트웨어인 R로 불러오고자 한다. 데이터를 불러오기 전에, 우선 데이터가 있는 경로를 설정해주어야 한다. 그 방법은 다음과 같다.

Results

```
> # 현재 경로 확인
> getwd()
[1] "C:/Users/SangmanJeong/Documents“

> # 새로운 경로 지정
> setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")

> # 현재 경로 재확인
> getwd()
[1] "C:/Users/SangmanJeong/Desktop/18년 2학기/통계학
```

R Code

```
# 현재 경로 확인
getwd()

# 새로운 경로 지정
setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")

# 현재 경로 재확인
getwd()
```

이 때, 경로 지정 시 \는 /로 바꾸어주고, 경로 전체를 “”로 묶어주어야 함을 주의하자. 이제 설정한 경로에 불러올 데이터를 넣고 이를 불러온다. 이제 다음과 같이 프로그래밍 한다.

Results

```
> # 불러온 데이터의 컬럼명 보기
> names(csdata)
[1] "obs"  "gpa"  "hsm"  "hss"  "hse"  "satm" "satv" "sex"

> # 데이터의 전반적인 기술 통계량 요약
> describe(csdata,skew=F)
     vars   n   mean     sd    min max  range   se
obs     1 224 112.50  64.81   1.00 224 223.00 4.33
gpa     2 224   2.64   0.78   0.12   4   3.88 0.05
hsm     3 224   8.32   1.64   2.00  10   8.00 0.11
hss     4 224   8.09   1.70   3.00  10   7.00 0.11
hse     5 224   8.09   1.51   3.00  10   7.00 0.10
satm    6 224 595.29  86.40 300.00 800 500.00 5.77
satv    7 224 504.55  92.61 285.00 760 475.00 6.19
sex     8 224   0.65   0.48   0.00   1   1.00 0.03
```

R Code

```
# txt 데이터 불러오기 (header는 컬럼 첫 번째 열 이름표로 사용 여부)
csdata <- read.table(file="csdata.txt",header=T)

# 불러온 데이터의 컬럼명 보기
names(csdata)

# psych 패키지 불러오기
library(psych)

# 데이터의 전반적인 기술 통계량 요약
describe(csdata,skew=F)
```

이제 table의 각 변수에 대한 분포의 이해를 돕기 위해 변수에 대한 빈도수, 확률, 누적 빈도, 누적 확률을 계산해본다. 다음과 같이 프로그래밍 한다.

Results

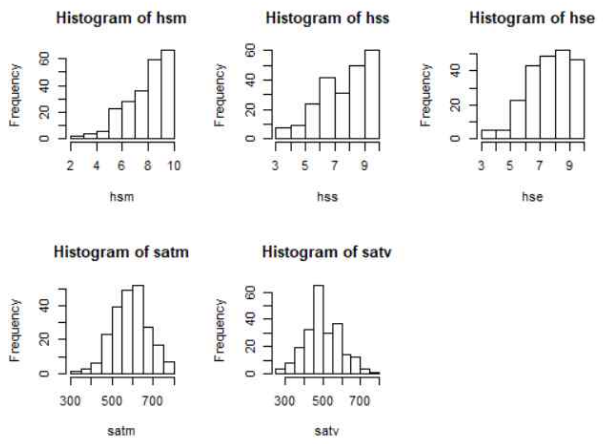
```
> # 변수에 대한 빈도수(명령어 : table())
> freqhsm=table(hsm)
> freqhsm
hsm
 2  3  4  5  6  7  8  9 10
 1  1  4  6 23 28 36 59 66

> # 변수에 대한 각각의 확률
> freqhsm/length(hsm)
hsm
          2           3           4           5           6
      7
0.004464286 0.004464286 0.017857143 0.026785714 0.102678571
0.125000000
          8           9          10
0.160714286 0.263392857 0.294642857

> # 변수에 대한 누적빈도수 (명령어 : cumsum())
> cumsum(freqhsm)
  2   3   4   5   6   7   8   9  10
  1   2   6  12  35  63  99 158 224

> # 변수에 대한 누적 확률
> cumsum(freqhsm)/length(hsm)
          2           3           4           5           6
      7
0.004464286 0.008928571 0.026785714 0.053571429 0.156250000
0.281250000
          8           9          10
0.441964286 0.705357143 1.000000000
```

R Code

```
# 데이터를 지정 (컬럼명 사용시 코드 간소화 위함)
attach(csdata)

# 변수에 대한 빈도수(명령어 : table())
freqhsm=table(hsm)
freqhsm

# 변수에 대한 각각의 확률
freqhsm/length(hsm)

# 변수에 대한 누적빈도수 (명령어 : cumsum())
cumsum(freqhsm)

# 변수에 대한 누적 확률
cumsum(freqhsm)/length(hsm)
```

각 변수에 대한 분포를 더욱 시각적으로 볼 수 있게 하는 방법 중 하나는 히스토그램이다. 히스토그램을 이용하여 각 변수의 분포를 살펴보도록 하자. 다음과 같이 입력하여라.

Results



R Code

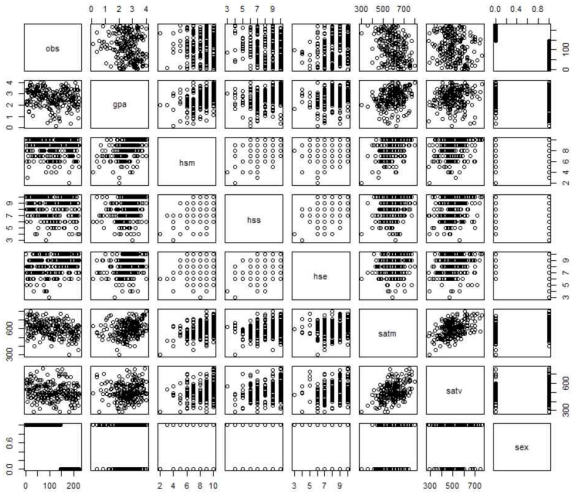
```
# 하나의 figure에 여러 그래프 나타내기
par(mfrow=c(2,3)) ## 2X3행렬 배열
hist(hsm)
hist(hss)
hist(hse)
hist(satm)
hist(satv)
```

satm 과 satv는 분포가 어느정도 정규성을 띄고 있음을 확인할 수 있다. 회귀분석을 실시하기 전에, 우선 상관관계에 대한 분석을 실시해본다. 상관관계는 상관계수를 구하여 그 정도를 비교해보면 된다.

우선 psych에 대한 패키지와 더불어 ltm 패키지를 설치하여 p-value도 함께 구해본다. 이는 상관계수가 0이라는 가설을 검정하기 위함이다. (이는 각 변수간 유의미한 상관이 있음을 검증하기 위한 절차이다.) 다음과 같이 프로그래밍을 실시한다.

Results

```
> # 상관계수에 대한 p-값 테스트
> rcor.test(csdata)

      obs     gpa     hsm     hss     hse     satm    satv    sex
obs   ***** -0.160  -0.064  -0.125   0.250  -0.314  -0.167  -0.828
gpa    0.016  *****   0.436   0.329   0.289   0.252   0.114  -0.048
hsm    0.341 <0.001   *****   0.576   0.447   0.454   0.221  -0.072
hss    0.062 <0.001  <0.001   *****   0.579   0.240   0.262  -0.011
hse   <0.001 <0.001  <0.001  <0.001   *****   0.108   0.244  -0.314
satm  <0.001 <0.001  <0.001  <0.001   0.106   *****   0.464   0.259
satv   0.012  0.087   0.001  <0.001  <0.001  <0.001   *****   0.063
sex   <0.001  0.476   0.283   0.873  <0.001  <0.001   0.348   *****

upper diagonal part contains correlation coefficient estimates
lower diagonal part contains corresponding p-values
```



R Code

```
# 각 변수간 p-값을 구하기 위한 라이브러리
library(ltm)

# 상관계수에 대한 p-값 테스트(상삼각:상관계수, 하삼각:p-값)
rcor.test(csdata)

# 모든 변수에 대한 상관분포
pairs(csdata)
```

이제, 회귀분석을 실시하자. 이 문제에서 회귀분석은 변수의 개수가 하나 이상이므로 다중 회귀분석을 실시하도록 한다. 그리고 회귀모형을 설정한 후 분산분석도 실시해보도록 한다. 코드는 다음과 같이 설계하면 된다.

Results

```
> # 회귀모형
> csfit=lm(gpa ~ satm+satv+hsm+hss+hse)
> summary(csfit)

Call:
lm(formula = gpa ~ satm + satv + hsm + hss + hse)

Residuals:
     Min       1Q   Median       3Q      Max
-2.06493 -0.30843  0.06894  0.48760  1.70543

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.3267187  0.3999964   0.817 0.414932
satm         0.0009436  0.0006857   1.376 0.170176
satv        -0.0004078  0.0005919  -0.689 0.491518
hsm          0.1459611  0.0392610   3.718 0.000256 ***
hss          0.0359053  0.0377984   0.950 0.343207
hse          0.0552926  0.0395687   1.397 0.163719
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7 on 218 degrees of freedom
Multiple R-squared:  0.2115,    Adjusted R-squared:  0.1934
F-statistic: 11.69 on 5 and 218 DF,  p-value: 5.058e-10

> aov(csfit)
Call:
   aov(formula = csfit)

Terms:
                     satm      satv      hsm      hss
hse Residuals
Sum of Squares    8.58293   0.00091  17.72647  1.37653
0.95680 106.81914
Deg. of Freedom         1         1         1        1
1       218

Residual standard error: 0.6999972
Estimated effects may be unbalanced
```

R Code

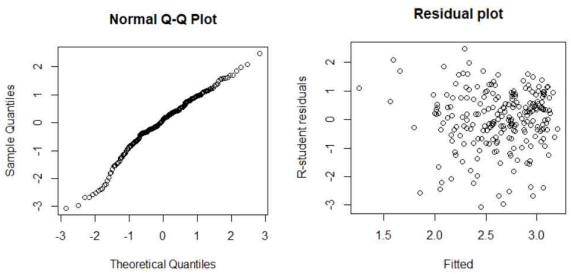
```
# 회귀모형
csfit=lm(gpa ~ satm+satv+hsm+hss+hse)
summary(csfit)
aov(csfit)
```

여기서 lm은 회귀모형을 만들어주는 명령어이고, gpa를 예측변수, ~ 오른쪽으로는 설명변수로 한다. 설명변수 개인가 여러 경우 + 기호로 연결해준다. summary는 구한 회귀모형의 전반적인 p-값과 검정통계량, 기술통계량 등을 확인할 수 있게 해준다.

분산분석을 실시하는 명령어는 aov()로 주어진다. 이 코드를 실행하게 되면 분산분석 TABLE이 나오고, 각 항에 대한 제곱합과 자유도, 표준오차 등을 확인할 수 있다.

구한 회귀모형에 대한 잔차 분석을 실시한다. 우선 잔차가 가져야 할 가정을 상기하자. 이는 정규성과 선형성 검정을 통해 확인할 수 있다. 다음의 코드를 입력한다.

Results



R Code

```
# 정규성 검정
t=rstudent(csfit)
qqnorm(t)

# 선형성 검정 (잔차 그래프)
y=csfit$fitted
plot(y,t,xlab='Fitted',ylab='R-student residuals',main="Residual plot")
```

▲

[friends.txt in R]

주어진 데이터 friends.txt에 대한 분산분석을 실시하고자 한다. 분산분석은 일원배치법(One-way)로 실시할 것이다. 우선 데이터를 살펴보기 위해 다음과 같은 과정을 실시한다.

Results

```
> # 현재 경로 확인
> getwd()
[1] "C:/Users/SangmanJeong/Documents"
>
> # 새로운 경로 지정
> setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")
>
> # 현재 경로 재확인
> getwd()
[1] "C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2"
>
> # txt 데이터 불러오기 (header는 컬럼 첫 번째 열 이름표로 사용 여부)
> frnddata <- read.table(file="friends.txt",header=T)
>
> # 불러온 데이터의 컬럼명 보기
> names(frnddata)
[1] "Friends"     "Participant" "Score"       "GroupNum"

> head(frnddata)
  Friends Participant Score GroupNum
1     102           1   3.8        1
2     102           2   3.6        1
3     102           3   3.2        1
4     102           4   2.4        1
5     102           5   4.8        1
6     102           6   3.0        1
> summary(frnddata)
     Friends       Participant        Score          GroupNum
 Min.   :102.0   Min.   :  1.00   Min.   :1.000   Min.   :1.000
 1st Qu.:302.0   1st Qu.: 34.25   1st Qu.:3.600   1st Qu.:2.000
 Median :502.0   Median : 67.50   Median :4.600   Median :3.000
 Mean   :488.6   Mean   : 67.50   Mean   :4.382   Mean   :2.933
 3rd Qu.:702.0   3rd Qu.:100.75   3rd Qu.:5.200   3rd Qu.:4.000
 Max.   :902.0   Max.   :134.00   Max.   :7.000   Max.   :5.000
```

R Code

```
# 현재 경로 확인
getwd()

# 새로운 경로 지정
setwd("C:/Users/SangmanJeong/Desktop/18년 2학기/통계학 2")

# 현재 경로 재확인
getwd()

# txt 데이터 불러오기 (header는 컬럼 첫 번째 열 이름표로 사용 여부)
frnddata <- read.table(file="friends.txt",header=T)

# 불러온 데이터의 컬럼명 보기
names(frnddata)

# 테이블 확인하기 및 데이터 요약
head(frnddata)
summary(frnddata)
```

데이터를 분산분석을 실시하기 위해 그룹별로 묶어줄 것이다. 친구 수에 따른 빈도수, 평균, 표준편차를 구한 테이블은 다음과 같이 만들 수 있다.

Results

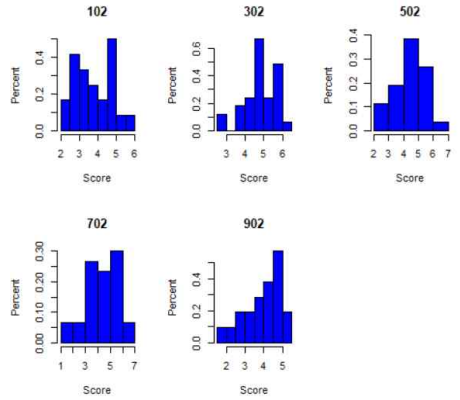
```
> ## 분산분석을 위한 데이터프레임

> friends
[1] 102 302 502 702 902
> n
[1] 24 34 26 30 21
> Mu
[1] 3.816667 4.876471 4.561538 4.406667 3.990476
> std
[1] 0.9989850 0.8384903 1.0703558 1.4282696 1.0226949

> frnddata2=data.frame(friends,n,Mu,std)
> frnddata2
  friends  n       Mu       std
1     102 24 3.816667 0.9989850
2     302 34 4.876471 0.8384903
3     502 26 4.561538 1.0703558
4     702 30 4.406667 1.4282696
5     902 21 3.990476 1.0226949
```

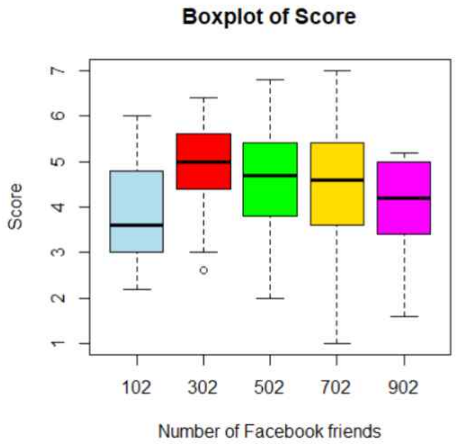
R Code

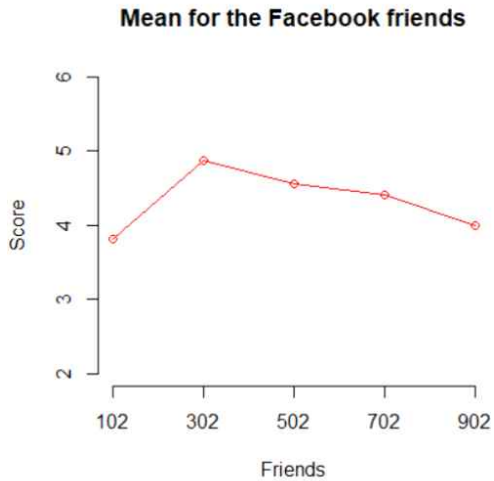
```
# 분산분석 테이블 만들기

## 항목
friends<-c(102,302,502,702,902)

## 항목별 표본크기
n1=length(frnddata$Friends[1:24])
n2=length(frnddata$Friends[24:57])
n3=length(frnddata$Friends[58:83])
n4=length(frnddata$Friends[84:113])
n5=length(frnddata$Friends[114:134])
n<-c(n1,n2,n3,n4,n5)

## 항목별 평균
Mu1=mean(frnddata$Score[1:24])
Mu2=mean(frnddata$Score[24:57])
Mu3=mean(frnddata$Score[58:83])
Mu4=mean(frnddata$Score[84:113])
Mu5=mean(frnddata$Score[114:134])
Mu<-c(Mu1,Mu2,Mu3,Mu4,Mu5)

## 항목별 표준편차
std1=sd(frnddata$Score[1:24])
std2=sd(frnddata$Score[24:57])
std3=sd(frnddata$Score[58:83])
std4=sd(frnddata$Score[84:113])
std5=sd(frnddata$Score[114:134])
std<-c(std1,std2,std3,std4,std5)

## 분산분석을 위한 데이터프레임
frnddata2=data.frame(friends,n,Mu,std)
frnddata2
```

분산분석을 실시하기 전에, 간단한 분포에 대한 정보를 시각적으로 확인할 필요가 있다. 히스토그램과 박스플롯, 꺾은선 그래프를 이용하여 확인해본다. 코드는 아래와 같이 작성한다.

Results







R Code

```
## 각 변수에 대한 히스토그램
attach(frnddata)
par(mfrow=c(2,3))
hist(Score[1:24],freq=FALSE,col="blue",xlab='Score',ylab='Percent',
main="102")
hist(Score[25:57],freq=FALSE,col="blue",xlab='Score',ylab='Percent'
,main="302")
hist(Score[58:83],freq=FALSE,col="blue",xlab='Score',ylab='Percent'
,main="502")
hist(Score[84:113],freq=FALSE,col="blue",xlab='Score',ylab='Percen
t',main="702")
hist(Score[114:134],freq=FALSE,col="blue",xlab='Score',ylab='Perce
nt',main="902")

## 각 변수에 대한 박스플롯
scbox<-c(Score[1:24],Score[25:57],Score[58:83],Score[84:113],Scor
e[114:134])
boxplot(scbox ~ Friends, main = "Boxplot of Score",
                xlab = "Number of Facebook friends",
                                        ylab =
"Score",col=c("lightblue","red","green","gold","magenta"))

## 평균에 대한 점과 꺾은선 그래프
plot(Mu,ylim=c(2,6),type="o",col=2,xlab='Friends',ylab='Score',
            main="Mean for the Facebook friends",axes=FALSE)
axis(1,at=1:5,lab=c("102","302","502","702","902"))
axis(2,at=2:6,lab=c(2,3,4,5,6))
```

이제 아래와 같이 프로그래밍하여 분산분석에 대한 결과를 제시하고 이를 분석해본다.

Results

```
> ## 일원배치 분산분석(ANOVA)
> analysis_friends = aov(scbox ~ Friends, data=frnddata)
> summary(analysis_friends)
             Df Sum Sq Mean Sq F value Pr(>F)
Friends       1    0.1  0.1015   0.077  0.782
Residuals   132  174.7  1.3231
```

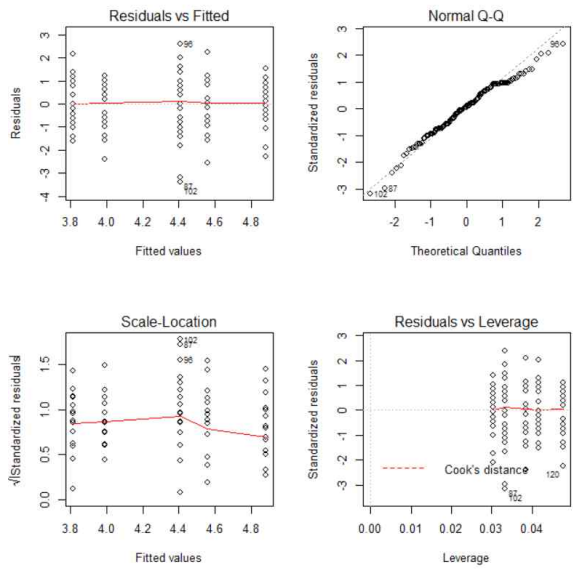
R Code

```
## 일원배치 분산분석(ANOVA)
analysis_friends = aov(scbox ~ Friends, data=frnddata)
summary(analysis_friends)
```

Results에 나타난 분산분석표의 결과를 보면, p-값이 0.782인 매우 큰 값을 가지므로, 어떠한 유의수준보다도 큰 값이다. 따라서 평균에 차이가 없다는 귀무가설을 기각할 수 없다. 즉, 페이스북 친구 수에 관한 매력 점수의 평균은 차이가 있다고 주장할 수 없다. 그런데 이는 이전의 박스 플롯과 꺾은선 그래프 등으로 살펴본 내용으로 짐작한 결과와는 다른 결과가 나왔다. 우선적으로, p-값이 매우 높게 나왔음은 심히 의심스럽다. 이는 Friends 컬럼의 데이터가 범주형 데이터(factor)가 아닌 정수 데이터 타입으로 지정되어 있기 때문에 벌어진 결과다. 이를 고친 후 다시 분산분석의 결과를 보도록 한다. 다음과 같이 코드를 입력한다.

Results

```
> ## 일원배치 분산분석(ANOVA)
> analysis_friends = aov(scbox ~ Friends, data=frnddata)
> summary(analysis_friends)
             Df Sum Sq Mean Sq F value   Pr(>F)
Friends       4  19.89   4.973   4.142 0.00344 **
Residuals   129 154.87   1.201
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```



R Code

```
# 테이블 주요 컬럼 범주형 데이터로 바꾸기
sapply(frnddata,class) ## 컬럼의 데이터 타입 확인
frnddata<-transform(frnddata,Friends=factor(Friends))  ##  데이터 타입 변경
sapply(frnddata,class)

## 일원배치 분산분석(ANOVA)
analysis_friends = aov(scbox ~ Friends, data=frnddata)
summary(analysis_friends)

## 분산분석 결과에 대한 플롯
par(mfrow=c(2,2))
plot(analysis_friends)
```

컬럼 타입을 범주형으로 바꾸고 나니 p-값이 0.00344로 구해졌다. 이는 유의수준 0.01에서 귀무가설이 기각됨을 말해준다. 따라서 우리가 예상한 바와 같이 페이스북 친구에 따른 평균의 차이가 있다고 주장할 수 있다. 소프트웨어를 이용할 때는 항상 숫자 값으로 지정되는 범주형 데이터에 대해 유의하도록 하자.

▲