

CH3_Regression_ex1

Philip oh

데이터 적재 및 확인

```
adv = read.csv("Advertising.csv", header=T, sep=",")
adv = adv[,-1] #관측치의 번호는 제거하였다.
names(adv) = tolower(names(adv)) #변수 이름을 소문자로 바꾸었다.
head(adv)
```

```
##      tv radio newspaper sales
## 1 230.1  37.8      69.2  22.1
## 2  44.5  39.3      45.1  10.4
## 3  17.2  45.9      69.3   9.3
## 4 151.5  41.3      58.5  18.5
## 5 180.8  10.8      58.4  12.9
## 6   8.7  48.9      75.0   7.2
```

모형의 적합

```
lm.fit = lm(sales ~ tv, data=adv)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = sales ~ tv, data = adv)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.3860 -1.9545 -0.1913  2.0671  7.2124
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.032594   0.457843   15.36  <2e-16 ***
## tv           0.047537   0.002691   17.67  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.259 on 198 degrees of freedom
## Multiple R-squared:  0.6119, Adjusted R-squared:  0.6099
## F-statistic: 312.1 on 1 and 198 DF, p-value: < 2.2e-16
```

- 다른 변수들이 고정되어 있지 않을 때 sales 변수에 대한 tv의 정보들을 확인할 수 있다.

회귀모델 안에 있는 객체들의 이름과 구조를 확인

```
names(lm.fit)
```

```
## [1] "coefficients" "residuals" "effects" "rank"
## [5] "fitted.values" "assign" "qr" "df.residual"
## [9] "xlevels" "call" "terms" "model"
```

```
str(lm.fit)
```

```

## List of 12
## $ coefficients : Named num [1:2] 7.0326 0.0475
## ..- attr(*, "names")= chr [1:2] "(Intercept)" "tv"
## $ residuals    : Named num [1:200] 4.13 1.25 1.45 4.27 -2.73 ...
## ..- attr(*, "names")= chr [1:200] "1" "2" "3" "4" ...
## $ effects      : Named num [1:200] -198.31 57.57 1.08 3.99 -2.98 ...
## ..- attr(*, "names")= chr [1:200] "(Intercept)" "tv" "" "" ...
## $ rank         : int 2
## $ fitted.values: Named num [1:200] 17.97 9.15 7.85 14.23 15.63 ...
## ..- attr(*, "names")= chr [1:200] "1" "2" "3" "4" ...
## $ assign       : int [1:2] 0 1
## $ qr          :List of 5
## ..$ qr        : num [1:200, 1:2] -14.1421 0.0707 0.0707 0.0707 0.0707 ...
## .. ..- attr(*, "dimnames")=List of 2
## .. .. ..$ : chr [1:200] "1" "2" "3" "4" ...
## .. .. ..$ : chr [1:2] "(Intercept)" "tv"
## .. ..- attr(*, "assign")= int [1:2] 0 1
## ..$ qraux: num [1:2] 1.07 1.09
## ..$ pivot: int [1:2] 1 2
## ..$ tol  : num 1e-07
## ..$ rank : int 2
## ..- attr(*, "class")= chr "qr"
## $ df.residual : int 198
## $ xlevels      : Named list()
## $ call         : language lm(formula = sales ~ tv, data = adv)
## $ terms        :Classes 'terms', 'formula' language sales ~ tv
## .. ..- attr(*, "variables")= language list(sales, tv)
## .. ..- attr(*, "factors")= int [1:2, 1] 0 1
## .. .. ..- attr(*, "dimnames")=List of 2
## .. .. .. ..$ : chr [1:2] "sales" "tv"
## .. .. .. ..$ : chr "tv"
## .. ..- attr(*, "term.labels")= chr "tv"
## .. ..- attr(*, "order")= int 1
## .. ..- attr(*, "intercept")= int 1
## .. ..- attr(*, "response")= int 1
## .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
## .. ..- attr(*, "predvars")= language list(sales, tv)
## .. ..- attr(*, "dataClasses")= Named chr [1:2] "numeric" "numeric"
## .. .. ..- attr(*, "names")= chr [1:2] "sales" "tv"
## $ model        :'data.frame': 200 obs. of 2 variables:
## ..$ sales: num [1:200] 22.1 10.4 9.3 18.5 12.9 7.2 11.8 13.2 4.8 10.6 ...
## ..$ tv : num [1:200] 230.1 44.5 17.2 151.5 180.8 ...
## ..- attr(*, "terms")=Classes 'terms', 'formula' language sales ~ tv
## .. .. ..- attr(*, "variables")= language list(sales, tv)
## .. .. ..- attr(*, "factors")= int [1:2, 1] 0 1
## .. .. .. ..- attr(*, "dimnames")=List of 2
## .. .. .. .. ..$ : chr [1:2] "sales" "tv"
## .. .. .. .. ..$ : chr "tv"
## .. .. ..- attr(*, "term.labels")= chr "tv"
## .. .. ..- attr(*, "order")= int 1
## .. .. ..- attr(*, "intercept")= int 1
## .. .. ..- attr(*, "response")= int 1
## .. .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
## .. .. ..- attr(*, "predvars")= language list(sales, tv)
## .. .. ..- attr(*, "dataClasses")= Named chr [1:2] "numeric" "numeric"
## .. .. .. ..- attr(*, "names")= chr [1:2] "sales" "tv"
## - attr(*, "class")= chr "lm"

```

회귀계수 추정량 확인

```
coef(lm.fit)
```

```
## (Intercept)          tv  
## 7.03259355  0.04753664
```

- `coef()` 를 통해 추정된 회귀계수를 확인할 수 있다. `lm.fit$coefficients` 와 동일한 기능을 한다.

```
confint(lm.fit, level=0.95)
```

```
##                2.5 %      97.5 %  
## (Intercept) 6.12971927 7.93546783  
## tv          0.04223072 0.05284256
```

- `confint()` 를 통해 신뢰구간을 확인할 수 있다. 기본값은 95% 신뢰구간이다.

신뢰구간 예측

```
predict(lm.fit, data.frame(tv=c(230.1, 44.5, 17.2)), interval="confidence")
```

```
##          fit          lwr          upr  
## 1 17.970775 17.337774 18.603775  
## 2  9.147974  8.439101  9.856848  
## 3  7.850224  7.024932  8.675515
```

- `predict()` 를 통해 신뢰구간(`interval="confidence"`)을 구했다.
- 첫 번째값을 보면, tv 광고예산이 230일 때 판매량이 17.9일 것이라 예측되었다.
- `lwr`은 신뢰구간의 하한값, `upr`은 상한값을 의미한다.
- 한 가지 주의해야 하는 것은, 이는 평균적인 차량에 대한 추정이므로 오차항은 고려하지 않았다. 선형회귀의 기본 가정에서 오차의 평균은 0이라고 가정하기 때문이다.

예측구간 예측

```
predict(lm.fit, data.frame(tv=c(230.1, 44.5, 17.2)), interval="prediction")
```

```
##          fit          lwr          upr  
## 1 17.970775 11.513546 24.42800  
## 2  9.147974  2.682867 15.61308  
## 3  7.850224  1.371318 14.32913
```

- `predcit()` 를 통해 예측구간(`interval="prediction"`)도 구할 수 있다.
- 예측값은 달라지지 않았다. 하지만, 신뢰구간에 비해 예측구간이 더 넓은 것을 확인할 수 있다. 이는 예측구간은 오차항을 포함하기 때문이다.

```
predict(lm.fit, data.frame(tv=c(230.1, 44.5, 17.2)), interval="none")
```

```
##          1          2          3
## 17.970775  9.147974  7.850224
```

- 만약 옵션 값을 `interval="none"` 이라고 한다면, 구간추정치를 계산하지 않는다. 예측치들만 나열된 것을 확인할 수 있다.