STA 4320 CHAP 3.1.3

He Jiang

Sep 2024

Sec 3.1.3

Advertising dataset

```
fpath = getwd()
Advertising = read.csv(paste0(fpath, "/Advertising.csv"))
x = Advertising$TV
y = Advertising$sales
n = nrow(Advertising)
```

```
res = summary( lm(y ~ x) )
res
```

```
##
## Call:
## lm(formula = y ~ x)
##
## 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 ***
## x           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
```

Computation of residual squared error

```
rse = sqrt(1 / (n-2) * sum( (y - predict(lm(y ~ x), new_obs = x) )^2 ) )
```

Computation of r squared

```
r_sq = 1 - ( sum( (y - predict(lm(y ~ x), new_obs = x))^2 ) / sum( (y - mean(y))^ 2) )
```

Simple point estimate

```
# note that the new value needs to be provided in data.frame
new_obs = data.frame(x = 200)
as.numeric( predict(lm(y ~ x), new_obs) )
```

```
## [1] 16.53992
```

Confidence interval (on the expectation of all locations' sales with a certain amount of TV spending)

```
new_obs = data.frame(x = 200)

predict(lm(y ~ x),
        new_obs,
        interval = "confidence",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 16.53992 16.00567 17.07418
```

Prediction interval (on one location's sale with a certain amount of TV spending)

```
new_obs = data.frame(x = 200)

predict(lm(y ~ x),
        new_obs,
        interval = "prediction",
        level = 0.95)
```

```
##        fit      lwr      upr
## 1 16.53992 10.09162 22.98822
```