homework1

Ashley Spirrison

2023-10-02

Creating the Data Set

```
gender <- c('M','M','F','M','F','F','M','F','M')
age <- c(34, 64, 38, 63, 40, 73, 27, 51, 47)
smoker <- c('no','yes','no','no','yes','no','no','no','yes')
exercise <- factor(c('moderate','frequent','some','some','moderate','none','none','moderate','moderate')
                   levels=c('none','some','moderate','frequent'), ordered=TRUE
)
los <- c(4,8,1,10,6,3,9,4,8)
x <- data.frame(gender, age, smoker, exercise, los)
x
```

```
##   gender age smoker exercise los
## 1      M  34     no moderate   4
## 2      M  64    yes frequent   8
## 3      F  38     no     some   1
## 4      M  63     no     some  10
## 5      F  40    yes moderate   6
## 6      F  73     no     none   3
## 7      M  27     no     none   9
## 8      F  51     no moderate   4
## 9      M  47    yes moderate   8
```

Creating the Model

```
lm(los ~ gender + age + smoker + exercise, dat=x)
```

```
##
## Call:
## lm(formula = los ~ gender + age + smoker + exercise, data = x)
##
## Coefficients:
## (Intercept)      genderM          age    smokeryes   exercise.L   exercise.Q
##    0.588144     4.508675     0.033377     2.966623    -2.749852    -0.710942
##  exercise.C
##    0.002393
```

Coefficient has the highest effect on 'los' is genderM

Creating a model using los and gender and assign it to the variable mod and running the summary function with mod as its argument.

```
gender <- c('M','M','F','M','F','F','M','F','M')
los <- c(4,8,1,10,6,3,9,4,8)
mod<-lm(los ~ gender,dat=x)
summary(mod)
```

```
##
## Call:
## lm(formula = los ~ gender, data = x)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -3.8   -0.5    0.2    1.2    2.5
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.500      1.099   3.186   0.0154 *
## genderM        4.300      1.474   2.917   0.0224 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.197 on 7 degrees of freedom
## Multiple R-squared:  0.5487, Adjusted R-squared:  0.4842
## F-statistic:  8.51 on 1 and 7 DF,  p-value: 0.02243
```

Determining the estimate for the intercept and gender using coef. funct.

```
coef(mod)
```

```
## (Intercept)     genderM
##         3.5         4.3
```

```
#intercept: 3.5, genderM: 4.3
```

Calculating the standard errors.

```
sqrt(diag(vcov(summary(mod))))
```

```
## (Intercept)     genderM
##    1.098701    1.474061
```

Calculating test statistics.

```r
mod <- lm(los ~ gender, dat=x)
mod.c <- coef(summary(mod))
mod.c[,1]/mod.c[,2]
```

```
## (Intercept)     genderM
##    3.185581    2.917110
```

Calculating the p value for gender.

```r
pt(mod.c[,1]/mod.c[,2],7,lower.tail=FALSE)*2
```

```
## (Intercept)     genderM
##  0.01537082  0.02243214
```

```r
#gender p-value:0.0224
```

Predicted Values generation.

```r
3.5+(x$gender=='M')*4.3
```

```
## [1] 7.8 7.8 3.5 7.8 3.5 3.5 7.8 3.5 7.8
```

Passing model to predict and fitted.

```r
predict(mod)
```

```
##   1   2   3   4   5   6   7   8   9
## 7.8 7.8 3.5 7.8 3.5 3.5 7.8 3.5 7.8
```

```r
fitted(mod)
```

```
##   1   2   3   4   5   6   7   8   9
## 7.8 7.8 3.5 7.8 3.5 3.5 7.8 3.5 7.8
```

Passing newdat to predict.

```r
newdat <- data.frame(gender=c('F','M','F'))
predict(mod,newdat)
```

```
##   1   2   3
## 3.5 7.8 3.5
```

Residuals determination.

```r
x$los-predict(mod)
```

```
##    1    2    3    4    5    6    7    8    9
## -3.8  0.2 -2.5  2.2  2.5 -0.5  1.2  0.5  0.2
```

Passing mod to residuals.

```r
residuals(mod)
```

```
##    1    2    3    4    5    6    7    8    9
## -3.8  0.2 -2.5  2.2  2.5 -0.5  1.2  0.5  0.2
```

Squaring residuals, summing, and comparing to passing mod to def.

```r
t<-residuals(mod)
sum(t^2)
```

```
## [1] 33.8
```

```r
deviance(mod)
```

```
## [1] 33.8
```

```r
#result in equivalent values
```

Passing mod to df. residual.

```r
df.residual(mod)
```

```
## [1] 7
```

Calculating SE.

```r
y<-deviance(mod)/df.residual(mod)
sqrt(y)
```

```
## [1] 2.197401
```

```
#standard error:2.197
```

###Noting it matches the output of the below line.

```
predict(mod, se.fit=TRUE)$residual.scale
```

```
## [1] 2.197401
```

Running t-test.

Creating subset of x where gender is M and assigning to men, doing same with women.

```
men<-subset(x,gender=='M')

women<-subset(x,gender=='F')
```

Calculating variance.

```
#men variance for los
var(men$los)
```

```
## [1] 5.2
```

```
#women variance for los
var(women$los)
```

```
## [1] 4.333333
```

```
#variance: men:5.2, women:4.33
```

Calling t-test function.

```
t.test(women$los,men$los)
```

```
##
##  Welch Two Sample t-test
##
## data:  women$los and men$los
## t = -2.9509, df = 6.8146, p-value = 0.02205
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -7.7647486 -0.8352514
## sample estimates:
## mean of x mean of y
##       3.5       7.8
```

```
t.test(women$los,men$los,var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  women$los and men$los
## t = -2.9171, df = 7, p-value = 0.02243
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -7.7856014 -0.8143986
## sample estimates:
## mean of x mean of y
##       3.5       7.8
```

```
#the second one (where var.equal=TRUE) matches the p value for gender from the model summary.
```

Alternative way to calling t-test.

```
t.test(los ~ gender, dat=x, var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  los by gender
## t = -2.9171, df = 7, p-value = 0.02243
## alternative hypothesis: true difference in means between group F and group M is not equal to 0
## 95 percent confidence interval:
##  -7.7856014 -0.8143986
## sample estimates:
## mean in group F mean in group M
##             3.5             7.8
```

```
# compare p-values
t.test(los ~ gender, dat=x, var.equal=TRUE)$p.value
```

```
## [1] 0.02243214
```

```
coef(summary(lm(los ~ gender, dat=x)))[2,4]
```

```
## [1] 0.02243214
```