HW1 Section 2

Philip Lee

9/9/2019

Problem 6

function

```
set.seed(1234571)  ##sets seed
x<-seq(-10,10,by=0.1)   ##creates x-values
B0<-3
B1<-0.2
sig<-1
get.yval<-function(Beta0,Beta1,sigma,x)
{
  sigma<-rnorm(length(x),mean=0,sd=sig)  ##creates error values
  predy<-Beta0+Beta1*x+sigma  ##equation
  return(predy)
}
y<-get.yval(Beta0=B0, Beta1=B1, sigma=sig,x=x) ##stores y-values from the function
result<-lm(y~x)    ##stores linear regression
```

Problem 6 part A

I would expect the intercept to be a positive value since the B1*x value at x=0 will be 0 and the error value will be somewhere between -1 and 1 since it is normally distributed with a mean 0 and standard deviation of 1. I'd expect the intercept to be maybe ranging from 2 to 4 as adding or subtracting 1 from B0 value of 3 will result in those values. The slope would probably be nearing 0 since the coefficient for slope is a small number that will not increase or decrease the response value to a significant amount for each interval of x.

Problem 6 part B

```
summary(result)
```

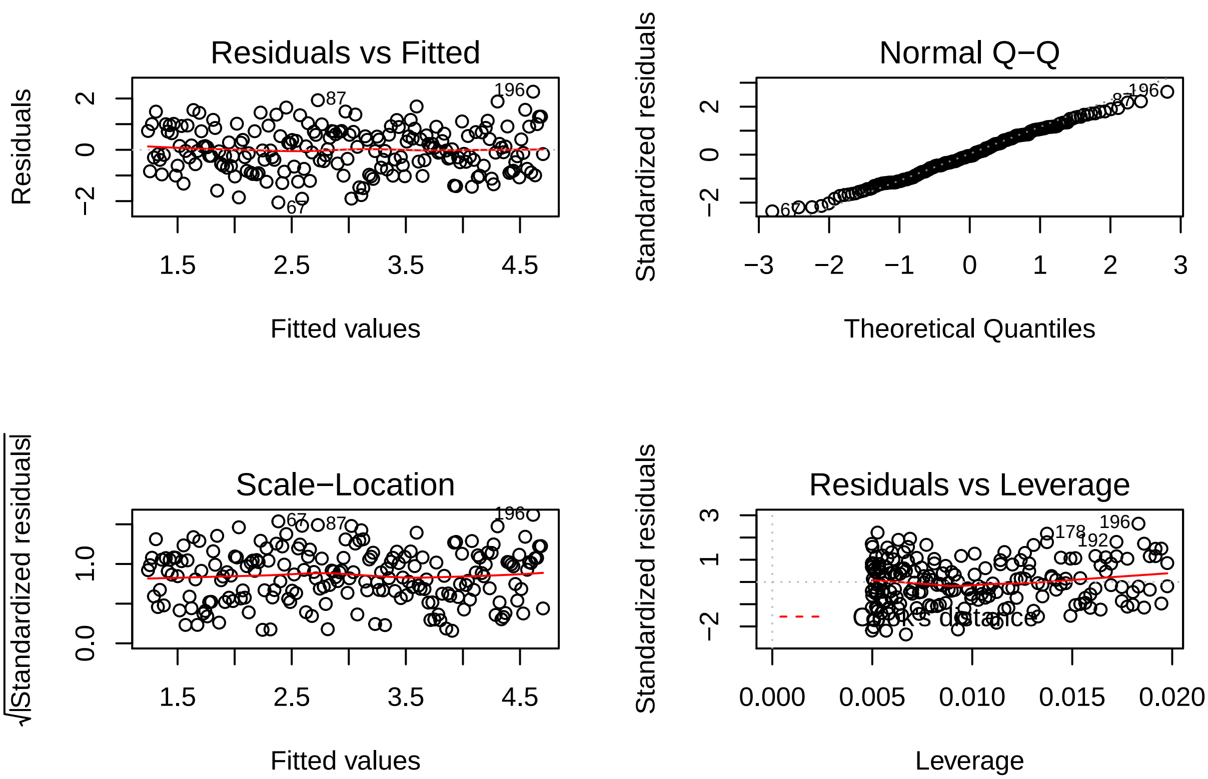
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.05228 -0.61898 -0.04652  0.69069  2.26452
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.97126    0.06149   48.32   <2e-16 ***
## x            0.17303    0.01060   16.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.8718 on 199 degrees of freedom
## Multiple R-squared:  0.5726, Adjusted R-squared:  0.5704
## F-statistic: 266.6 on 1 and 199 DF,  p-value: < 2.2e-16
```

Intercept: 2.9713. Slope: 0.1730. These values verify that the function is written correctly since the intercept and the slope were similar to how I expected them to be. In addition, these values are close in values to the known parameters of B0 and B1 which are 3 and 0.2 respectively, which further verifies that my function was written correctly.

Problem 6 part C

```r
par(mfrow=c(2,2))
plot(result)
```



Since the function I wrote was simulating a linear model, the residual graph shows that the function correctly simulated the linear model as the residual plot has the red line horizontal along the x-axis without any visible curvature, which indiates that the model is a good fit for a linear model as the residuals are scattered evenly throughout in a random pattern.

Problem 7

Problem 7 part A

```r
sig=1
get.yval2<-function(X)
{
```

```
  sigma<-rnorm(length(X),mean=0,sd=sig)   ##creates the error value
  intercept<-sin(X)
  predy<-intercept+0.2*X+sigma    ##gets the y-value
  return(predy)
}
```

Problem 7 part B

```
set.seed(1234571)  ##sets seed
reps<-10000       ##sets repetition amount
store.predy<-array(0,reps)    ##makes an array to hold predicted y-values
store.truey<-array(0,reps)  ##makes an array to hold true y-values
for (i in 1:reps){
  X<-seq(-10,10,by=0.1)     ##makes x-values
  yvals2<-get.yval2(X)     ##gets simulated y-values
  lsr<-lm(yvals2~X)      ##gets least sqaures regression with simulated data
  x0<-7    ##x0 value
  y0<-as.numeric(lsr$coefficients[1]+lsr$coefficients[2]*x0)   ##gets predicted y-value at x0=7
  trueY<-get.yval2(x0)      ##gets the true y-value at x0=7
  store.predy[i]<-y0     ##stores predicted y-value
  store.truey[i]<-trueY     ##stores true y-value
}
```

Problem 7 part C

```
mean((store.truey-store.predy)^2) ##gets test MSE
```

```
## [1] 1.278808
```

Problem 7 part D

```
avgpredy<-mean(store.predy)     ##gets avg predicted y-value at x0=7
print(avgpredy)
```

```
## [1] 1.556039
```

Problem 7 part E

```
f_x<-(sin(x0)+0.2*x0)     ##gets f(x0) value
var_fhat<-mean((store.predy-avgpredy)^2)     ##first part of the equation
print(var_fhat)
```

```
## [1] 0.01242472
```

```
bias<-(f_x-avgpredy)^2     ##second part of the equation
print(bias)
```

```
## [1] 0.2509484
```

```
var.e<-mean((store.truey-f_x)^2)  ##third part of the equation
print(var.e)
```

```
## [1] 1.012711
```

```r
var_fhat+bias+var.e  ##sum of the errors
```

```
## [1] 1.276084
```

Problem 7 part F

The sum of the three sources of error are similar to the MSE calculated in part C, which confirms I have successfully decomposed the test MSE into the three sources of error.