

# HW4

Jon Griffith and Lauren Quesada

2025-04-07

10

```
library(ISLR2)
```

```
## Warning: package 'ISLR2' was built under R version 4.4.3
```

(a)

```
df <- Carseats

fit <- lm(Sales ~ Price + Urban + US, data=df)
summary(fit)

##
## Call:
## lm(formula = Sales ~ Price + Urban + US, data = df)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -6.9206 -1.6220 -0.0564  1.5786  7.0581 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 13.043469  0.651012 20.036 < 2e-16 ***
## Price       -0.054459  0.005242 -10.389 < 2e-16 ***
## UrbanYes    -0.021916  0.271650 -0.081   0.936  
## USYes       1.200573  0.259042  4.635  4.86e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.472 on 396 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2335 
## F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16
```

(b)

The coefficient for 'Price' can be interpreted as for each dollar increase in Price, you can expect the number of carseats sold to go down by approximately 54 units.

The coefficient for ‘UrbanYes’ can be interpreted as stores located in urban areas sell, on average, approximately 22 fewer units than stores located in rural areas.

The coefficient for ‘USYes’ can be interpreted as stores located in the US sell, on average, approximately 1200 more units than stores not located in the US.

The ‘Price’ and ‘US’ variables are both statistically significant while the ‘Urban’ variable is not.

(c)

$$Sales = 13.043 - 0.054(\text{Price}) - 0.022(I_{\text{Urban}=\text{'Yes}}) + 1.2(I_{\text{US}=\text{'Yes}})$$

(d)

We can reject the null hypothesis for the ‘Urban’ variable.

(e)

```
fit2 <- lm(Sales ~ Price + US, data = df)
summary(fit2)

##
## Call:
## lm(formula = Sales ~ Price + US, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -6.9269 -1.6286 -0.0574  1.5766  7.0515 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 13.03079   0.63098  20.652 < 2e-16 ***
## Price       -0.05448   0.00523 -10.416 < 2e-16 ***
## USYes        1.19964   0.25846   4.641 4.71e-06 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.469 on 397 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2354 
## F-statistic: 62.43 on 2 and 397 DF,  p-value: < 2.2e-16
```

(f)

The first model has an adjusted R-squared of 0.2335 while the second model has an adjusted R-squared of 0.2354, suggesting a better fit for the second model with one less variable. This also means that each model accounts for about 23.54 percent of variation in the predictions for sales. Whether this is a good or bad fit is subjective, but it seems like a poor fit since the goal here is presumably to make business decisions based on predicted unit sales. If these predictions aren’t very accurate, you would not want to make important decisions based on these models.

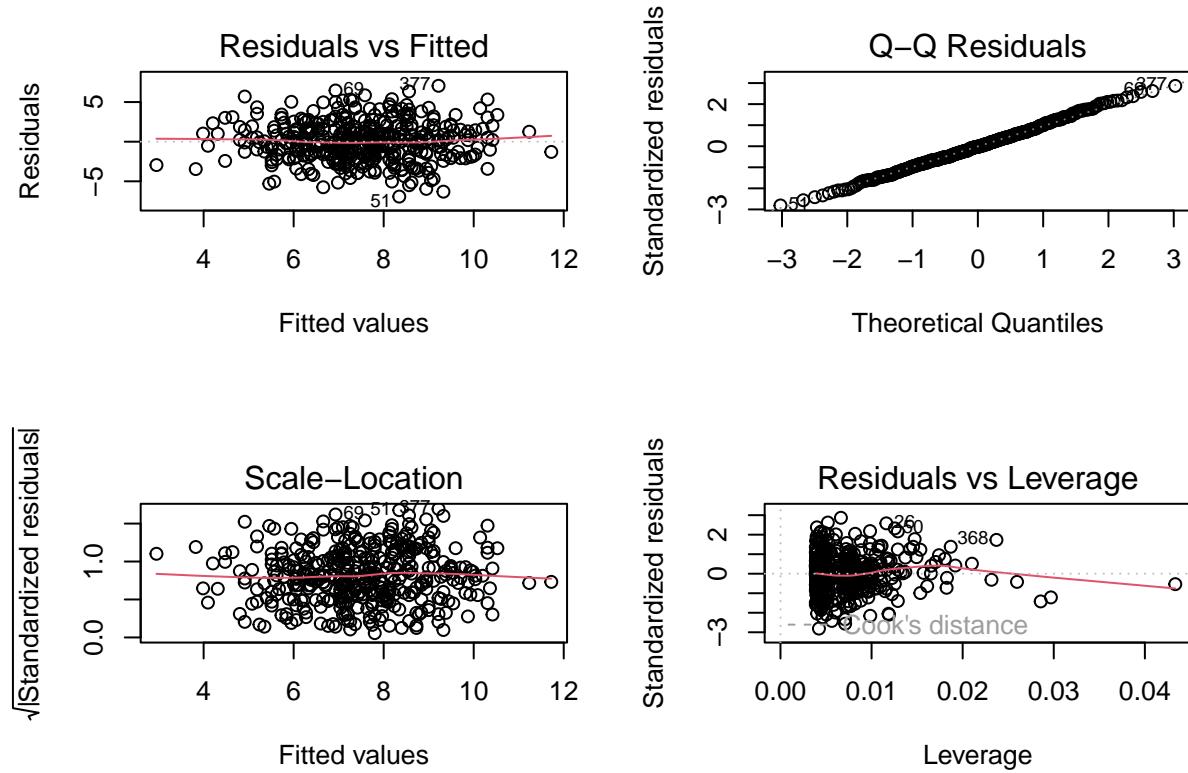
(g)

```
confint(fit2, level=0.95)

##              2.5 %      97.5 %
## (Intercept) 11.79032020 14.27126531
## Price       -0.06475984 -0.04419543
## USYes        0.69151957  1.70776632
```

(h)

```
par(mfrow=c(2,2))
plot(fit2)
```



```
#plot(fit2, which=5)
```

Based on the residual plots above, there does not appear to be any outliers or high leverage points in the dataset. There are a few points in the Residuals vs Leverage plot that appear to have high leverage relative to the rest of the points, but do not appear to be a big problem according to Cook's distance. All points along the QQ plot suggest a normal distribution with no sign of outliers.

13

(a) - (c)

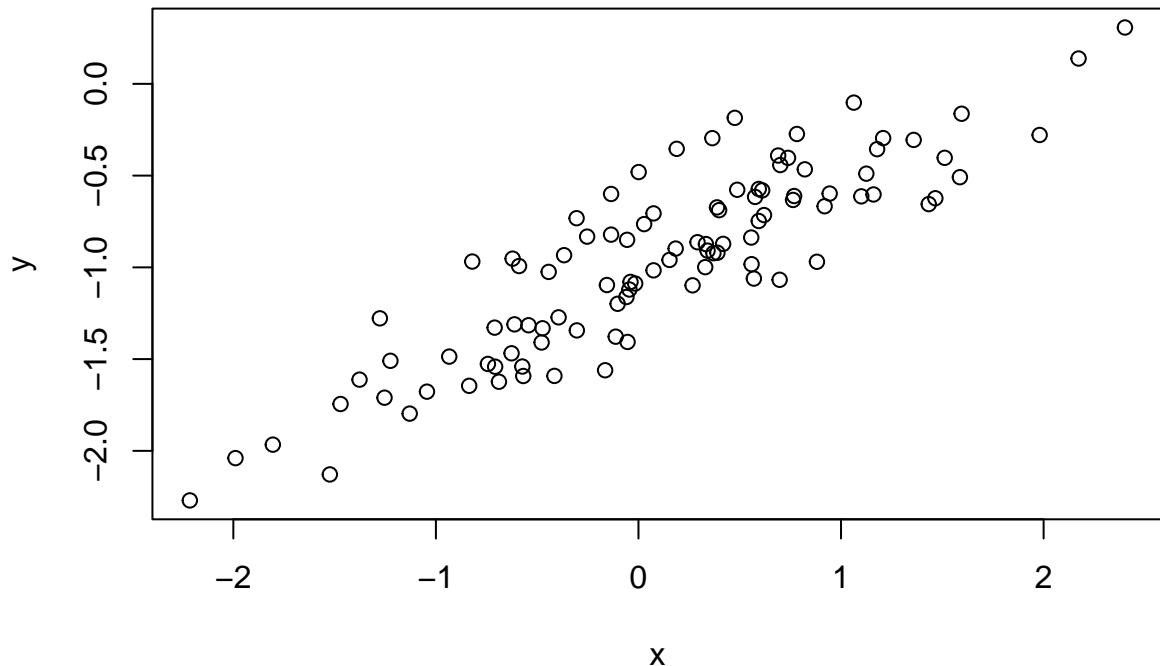
```
set.seed(1)

x <- rnorm(100)
eps <- rnorm(100, 0, sd=0.25)
y <- -1 + 0.5*x + eps
```

Vector 'y' has a length of 100. For this model,  $\beta_0 = -1$  and  $\beta_1 = 0.5$ .

(d)

```
plot(x, y)
```



We observe a positive linear relationship between x and y, which accurately reflects our ground truth of both variables.

(e)

```

fit_e <- lm(y ~ x)
summary(fit_e)

##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46921 -0.15344 -0.03487  0.13485  0.58654
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.00942   0.02425 -41.63   <2e-16 ***
## x            0.49973   0.02693  18.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2407 on 98 degrees of freedom
## Multiple R-squared:  0.7784, Adjusted R-squared:  0.7762
## F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16

```

The model has a statistically significant  $\hat{\beta}_0 = -1.009$  for the intercept and  $\hat{\beta}_1 = 0.4997$  for ‘x’, which accurately reflects the true coefficients of  $\beta_0 = -1$  and  $\beta_1 = 0.5$ . We also see that the R-squared is approximately 0.7784 which suggests that ‘x’ accounts for approximately 77.8 percent of the explained variance for the predictions of ‘y’. This also accurately reflects that our error term ‘eps’ has a variance of 0.25.

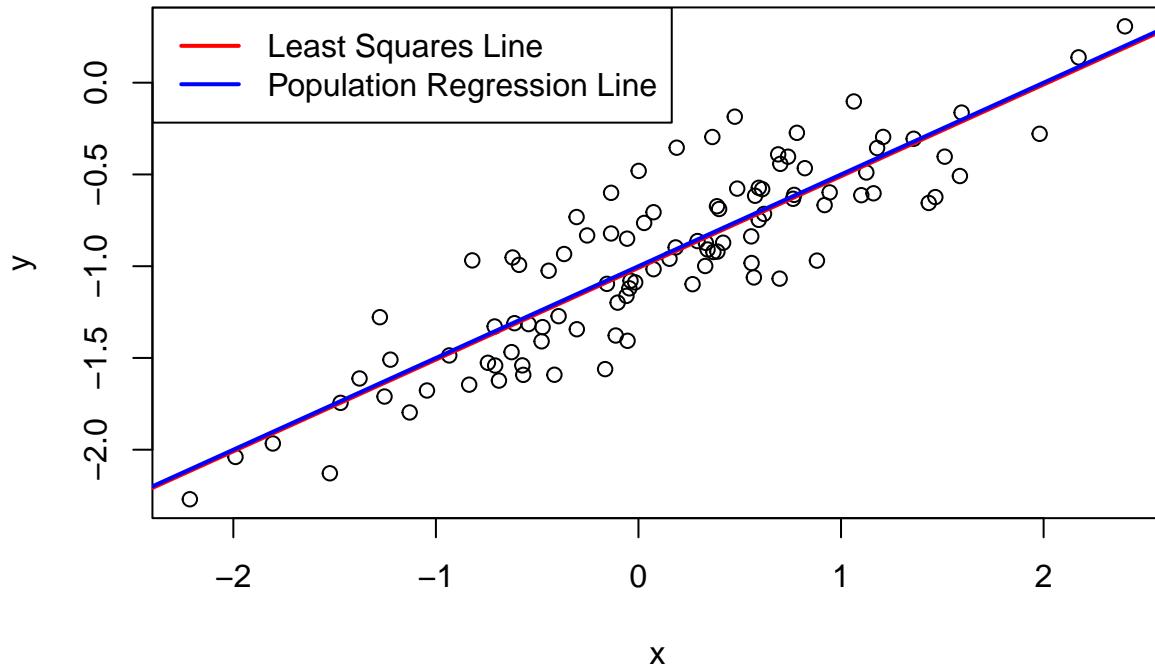
(f)

```

plot(x,y,
      main='Least Squares Line vs Population Regression Line')
abline(fit_e, col='red', lwd=2)
abline(a=-1, b=0.5, col='blue', lwd=2)
legend('topleft', c('Least Squares Line', 'Population Regression Line'), col=c('red', 'blue'), lwd=2)

```

## Least Squares Line vs Population Regression Line



(g)

```
fit4 <- lm(y ~ x + I(x^2))
summary(fit4)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.4913 -0.1563 -0.0322  0.1451  0.5675 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -0.98582   0.02941 -33.516  <2e-16 ***
## x            0.50429   0.02700  18.680  <2e-16 ***
## I(x^2)      -0.02973   0.02119  -1.403   0.164    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2395 on 97 degrees of freedom
## Multiple R-squared:  0.7828, Adjusted R-squared:  0.7784 
## F-statistic: 174.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

There is no evidence that the quadratic term improves the fit of the model. We conclude this based on a high p-value that doesn't beat any standard threshold, therefore we fail to reject the null hypothesis  $H_0 : \beta_2 = 0$ . We also only observe a marginal increase in the adjusted R-squared as further evidence that this does not do much to improve the fit. This makes sense since the true equation is not quadratic.

(h)

```

set.seed(1)

x <- rnorm(100)
eps <- rnorm(100, 0, sd=0.125)
y <- -1 + 0.5*x + eps

fit_h <- lm(y ~ x)

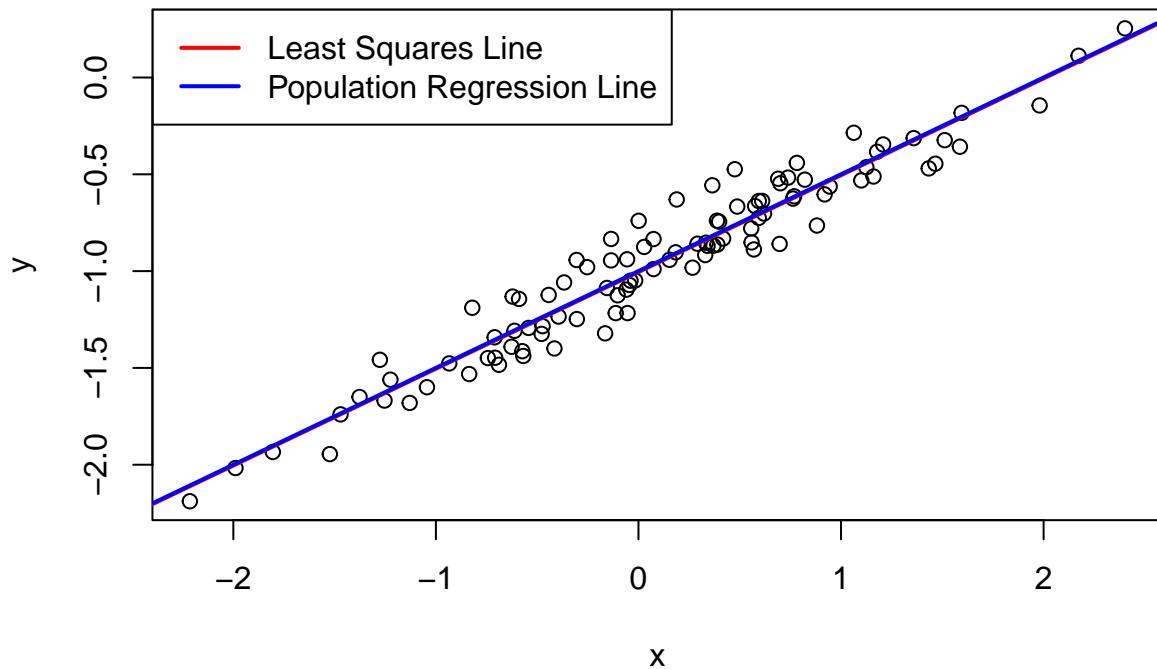
summary(fit_h)

##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23461 -0.07672 -0.01744  0.06742  0.29327
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.00471    0.01212  -82.87  <2e-16 ***
## x            0.49987    0.01347   37.12  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1203 on 98 degrees of freedom
## Multiple R-squared:  0.9336, Adjusted R-squared:  0.9329
## F-statistic: 1378 on 1 and 98 DF,  p-value: < 2.2e-16

plot(x,y,
      main='Least Squares Line vs Population Regression Line')
abline(fit_h, col='red', lwd=2)
abline(a=-1, b=0.5, col='blue', lwd=2)
legend('topleft', c('Least Squares Line', 'Population Regression Line'), col=c('red', 'blue'), lwd=2)

```

## Least Squares Line vs Population Regression Line



We see that we have similar values for both  $\hat{\beta}_0$  and  $\hat{\beta}_1$  which very closely approximate their true values. The variation is now even more explained by this model with an increase in proportion comparable to the decrease in variance for the true function. From the plot, we see that the OLS line is an even better approximation of the population regression line than we saw in the previous model, which makes sense since the data points are tighter around the line due to lower random error.

(i)

```
set.seed(1)

x <- rnorm(100)
eps <- rnorm(100, 0, sd=0.5)
y <- -1 + 0.5*x + eps

fit_i <- lm(y ~ x)

summary(fit_i)

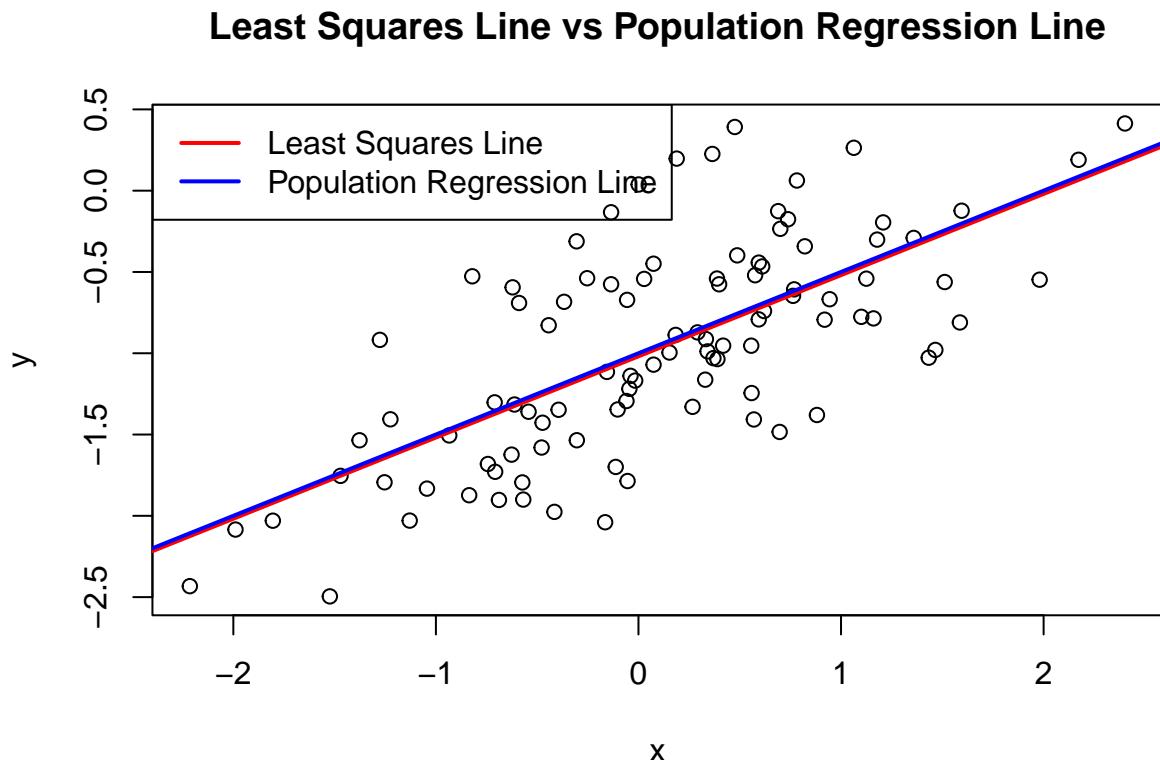
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.0000 -0.4875 -0.0250  0.4875  2.0000
```

```

## -0.93842 -0.30688 -0.06975  0.26970  1.17309
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.01885    0.04849 -21.010 < 2e-16 ***
## x           0.49947    0.05386   9.273 4.58e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4814 on 98 degrees of freedom
## Multiple R-squared:  0.4674, Adjusted R-squared:  0.4619
## F-statistic: 85.99 on 1 and 98 DF,  p-value: 4.583e-15

plot(x,y,
      main='Least Squares Line vs Population Regression Line')
abline(fit_i, col='red', lwd=2)
abline(a=-1, b=0.5, col='blue', lwd=2)
legend('topleft', c('Least Squares Line', 'Population Regression Line'), col=c('red', 'blue'), lwd=2)

```



We once again see similar coefficient estimates for  $\hat{\beta}_0$  and  $\hat{\beta}_1$  as the previous two models, which closely approximates the true values. We also see a comparable decrease in  $R^2$  proportional to variance increase in the error term. As the error term variance goes up to 0.5, we see that we now only explain approximately 46.75 percent of variance which is close to 50 percent. For the plot, we now see that the lines can be differentiated moreso than all previous models meaning that the OLS fit is slightly less accurate (though still very accurate) than the previous models. This makes sense with the increase in the variance for the error term resulting in a larger spread of data points.

```
cat("Second fit with e ~ N(0, 0.125) \n")
```

```
## Second fit with e ~ N(0, 0.125)
```

```
confint(fit_h)
```

```
##           2.5 %    97.5 %
## (Intercept) -1.0287701 -0.9806531
## x           0.4731449  0.5265901
```

```
cat('\n')
```

```
cat("First fit with e ~ N(0, 0.25) \n")
```

```
## First fit with e ~ N(0, 0.25)
```

```
confint(fit_e)
```

```
##           2.5 %    97.5 %
## (Intercept) -1.0575402 -0.9613061
## x           0.4462897  0.5531801
```

```
cat('\n')
```

```
cat("Third fit with e ~ N(0, 0.5) \n")
```

```
## Third fit with e ~ N(0, 0.5)
```

```
confint(fit_i)
```

```
##           2.5 %    97.5 %
## (Intercept) -1.1150804 -0.9226122
## x           0.3925794  0.6063602
```

We put the models in ascending order in terms of variance for the error term, with the second fit first, the original fit second, and the third fit third. We see that there is a positive correlation with variance in the error term and width of the confidence interval. That is, as the data set becomes more noisy, the range of values in the confidence interval becomes wider. This makes sense since the proportion of the variance explained by the model is also going down, meaning we become less confident in our ability to explain the variance in the response variable  $y$ , which also corresponds to a wider range of values for our confidence interval.

## 14

(a)

```

set.seed(1)
x1 <- runif(100)
x2 <- 0.5*x1 + rnorm(100) / 10
y <- 2 + 2*x1 + 0.3*x2 + rnorm(100)

```

The linear model has the form

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon \sim N(0, 1)$$

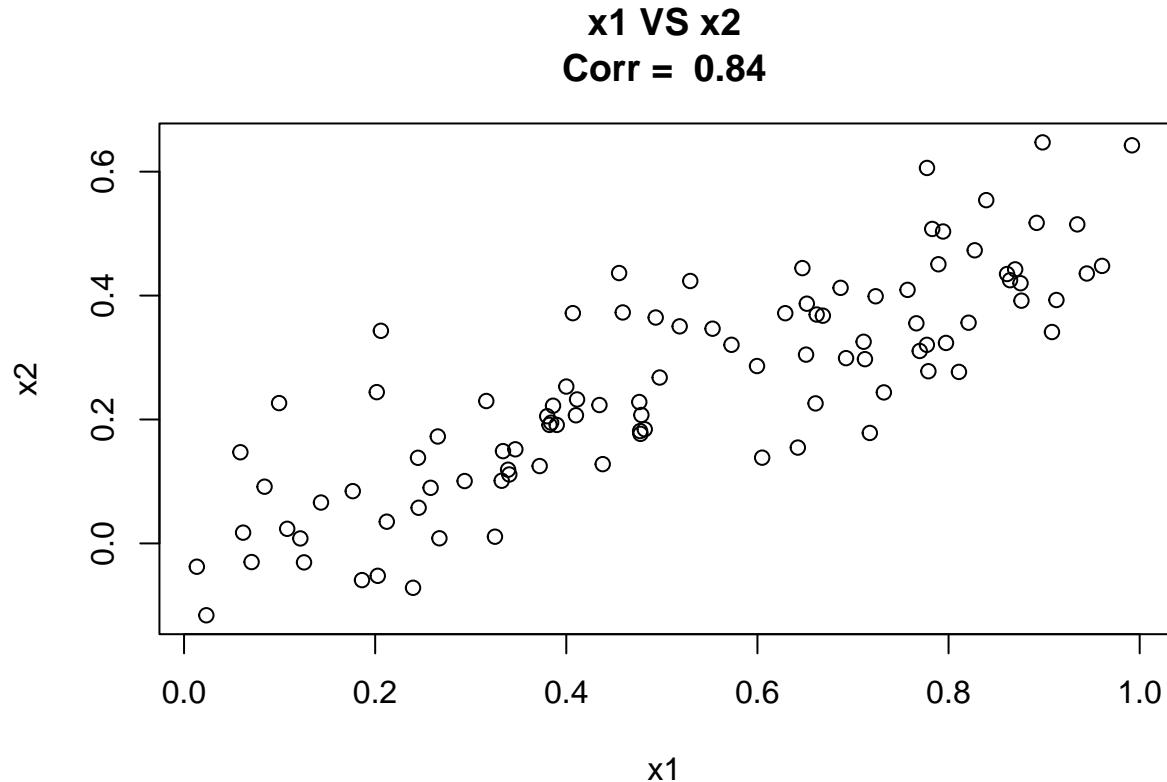
$$Y = 2 + 2X_1 + 0.3X_2 + \epsilon$$

(b)

```

plot(x1,x2,
      main = paste('x1 VS x2\nCorr = ', round(cor(x1,x2),2)))

```



(c)

```

fit_14c <- lm(y ~ x1 + x2)
summary(fit_14c)

```

```

## 
## Call:
## lm(formula = y ~ x1 + x2)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.1305    0.2319   9.188 7.61e-15 ***
## x1          1.4396    0.7212   1.996  0.0487 *  
## x2          1.0097    1.1337   0.891  0.3754    
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925 
## F-statistic: 12.8 on 2 and 97 DF,  p-value: 1.164e-05

```

The results of this model show that  $X_1$  is a statistically significant predictor of  $Y$ , but  $X_2$  is not. This means we can reject the null hypothesis for  $\beta_1$  but we can't reject the null hypothesis for  $\beta_2$ . We also see that only about 20 percent of the variance in the predictions are explained by these variables, based on  $R^2$ .

The coefficients are  $\beta_0 = 2.13$ ,  $\beta_1 = 1.4396$ , and  $\beta_2 = 1.0097$ . The intercept estimate is accurate while the coefficients for the variables are not, which suggests something is going on (collinearity).

(d)

```

fit_14d <- lm(y ~ x1)
summary(fit_14d)

## 
## Call:
## lm(formula = y ~ x1)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89495 -0.66874 -0.07785  0.59221  2.45560
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.1124    0.2307   9.155 8.27e-15 ***
## x1          1.9759    0.3963   4.986 2.66e-06 *** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.055 on 98 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.1942 
## F-statistic: 24.86 on 1 and 98 DF,  p-value: 2.661e-06

```

This fit has an even lower p-value for  $X_2$  showing that we can still reject the null hypothesis, but this time with a coefficient that approximately matches the true  $\beta_1$ . It makes sense that the coefficient estimate would match the true value in this case since  $X_1$  is independent of  $X_2$ .

(e)

```
fit_14e <- lm(y ~ x2)
summary(fit_14e)

##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.3899     0.1949   12.26 < 2e-16 ***
## x2          2.8996     0.6330    4.58 1.37e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.072 on 98 degrees of freedom
## Multiple R-squared:  0.1763, Adjusted R-squared:  0.1679
## F-statistic: 20.98 on 1 and 98 DF,  p-value: 1.366e-05
```

We see that  $X_2$  is now statistically significant meaning we can reject the null that  $\beta_2 = 0$ . However, the coefficient estimate for  $X_2$  does not come close to the true value for  $\beta_2$ . This makes sense since it is highly correlated with and depends on  $X_1$ . It is therefore trying to signal the magnitude of the  $X_1$  variable's impact on  $Y$  through  $X_2$  since  $X_1$  wasn't included in the model.

(f)

The results in (c)-(e) don't contradict each other since there is a high correlation between  $X_1$  and  $X_2$ . When we include both in the model, we see that  $X_1$  impact is being underestimated while  $X_2$  is being overestimated, which can be attributed to collinearity. The second model has just  $X_1$  which is independent of  $X_2$ , so we get its isolated impact on the response pretty accurately. The third model with just  $X_2$  is statistically significant, but mainly because it is essentially a proxy for both the impact of  $X_1$  and itself.

```
fit_14f <- lm(y ~ x1*x2)
summary(fit_14f)
```

```
##
## Call:
## lm(formula = y ~ x1 * x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
```

```

## -2.79572 -0.67391 -0.05085  0.61296  2.29607
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.2418     0.3167   7.079 2.39e-10 ***
## x1          1.1235     0.9467   1.187    0.238
## x2          0.3833     1.6603   0.231    0.818
## x1:x2      1.2497     2.4121   0.518    0.606
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.06 on 96 degrees of freedom
## Multiple R-squared:  0.211, Adjusted R-squared:  0.1864
## F-statistic: 8.559 on 3 and 96 DF,  p-value: 4.296e-05

```

(g)

```

x1 <- c(x1, 0.1)
x2 <- c(x2, 0.8)
y <- c(y, 6)

fitg1 <- lm(y~x1 + x2)
fitg2 <- lm(y~x1)
fitg3 <- lm(y~x2)

```

```
summary(fitg1)
```

```

##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.73348 -0.69318 -0.05263  0.66385  2.30619
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.2267     0.2314   9.624 7.91e-16 ***
## x1          0.5394     0.5922   0.911  0.36458
## x2          2.5146     0.8977   2.801  0.00614 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.075 on 98 degrees of freedom
## Multiple R-squared:  0.2188, Adjusted R-squared:  0.2029
## F-statistic: 13.72 on 2 and 98 DF,  p-value: 5.564e-06

```

```
summary(fitg2)
```

```

##
## Call:
```

```

## lm(formula = y ~ x1)
##
## Residuals:
##   Min     1Q Median     3Q    Max
## -2.8897 -0.6556 -0.0909  0.5682  3.5665
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.2569    0.2390  9.445 1.78e-15 ***
## x1          1.7657    0.4124  4.282 4.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.111 on 99 degrees of freedom
## Multiple R-squared:  0.1562, Adjusted R-squared:  0.1477
## F-statistic: 18.33 on 1 and 99 DF,  p-value: 4.295e-05

```

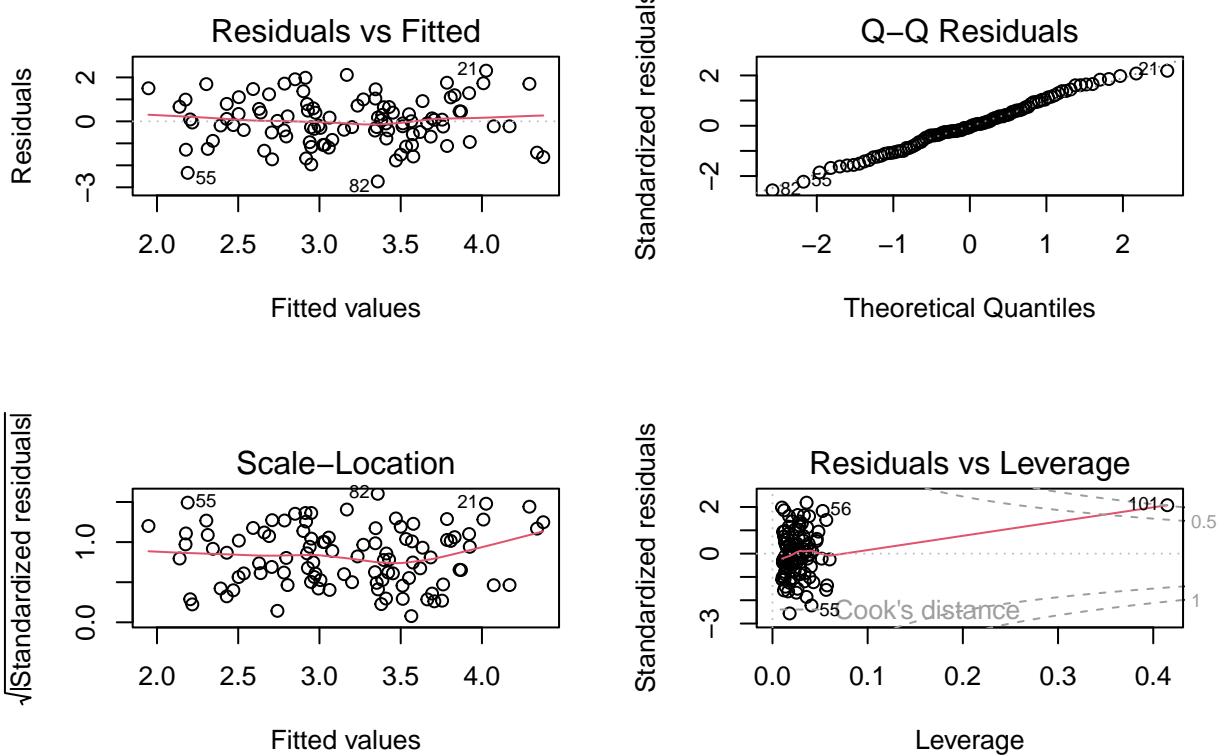
```
summary(fitg3)
```

```

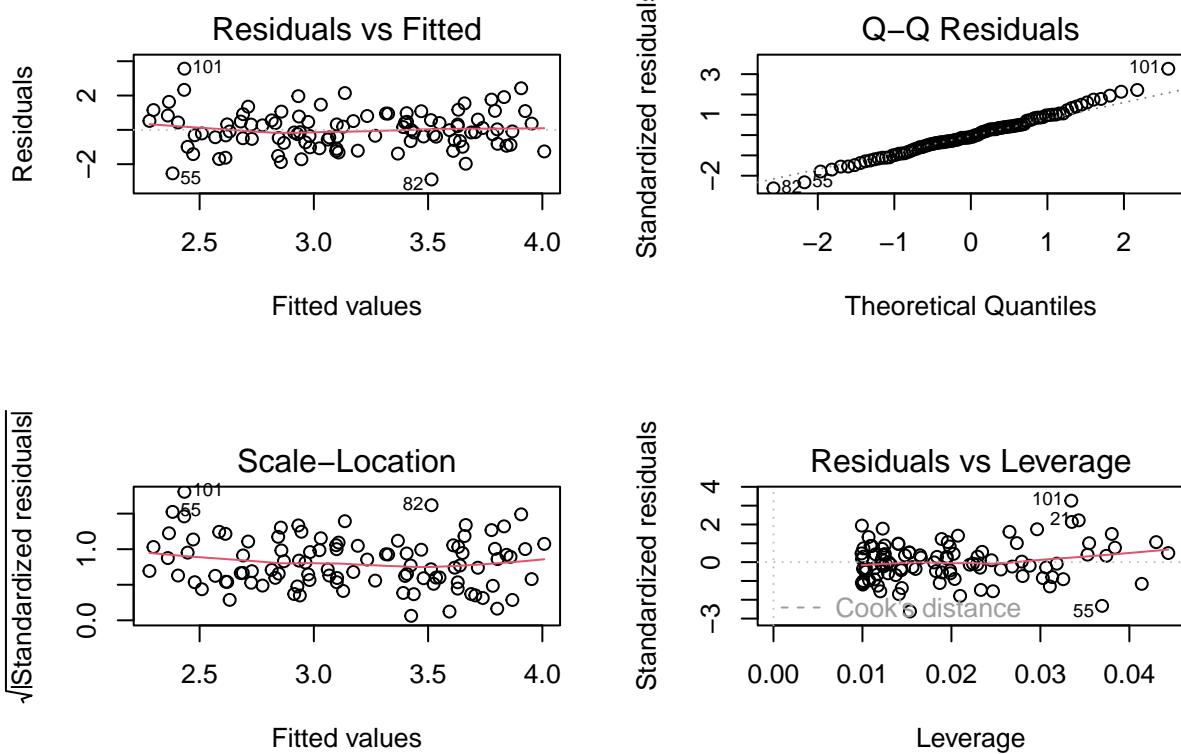
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##   Min     1Q Median     3Q    Max
## -2.64729 -0.71021 -0.06899  0.72699  2.38074
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.3451    0.1912 12.264 < 2e-16 ***
## x2          3.1190    0.6040  5.164 1.25e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.074 on 99 degrees of freedom
## Multiple R-squared:  0.2122, Adjusted R-squared:  0.2042
## F-statistic: 26.66 on 1 and 99 DF,  p-value: 1.253e-06

```

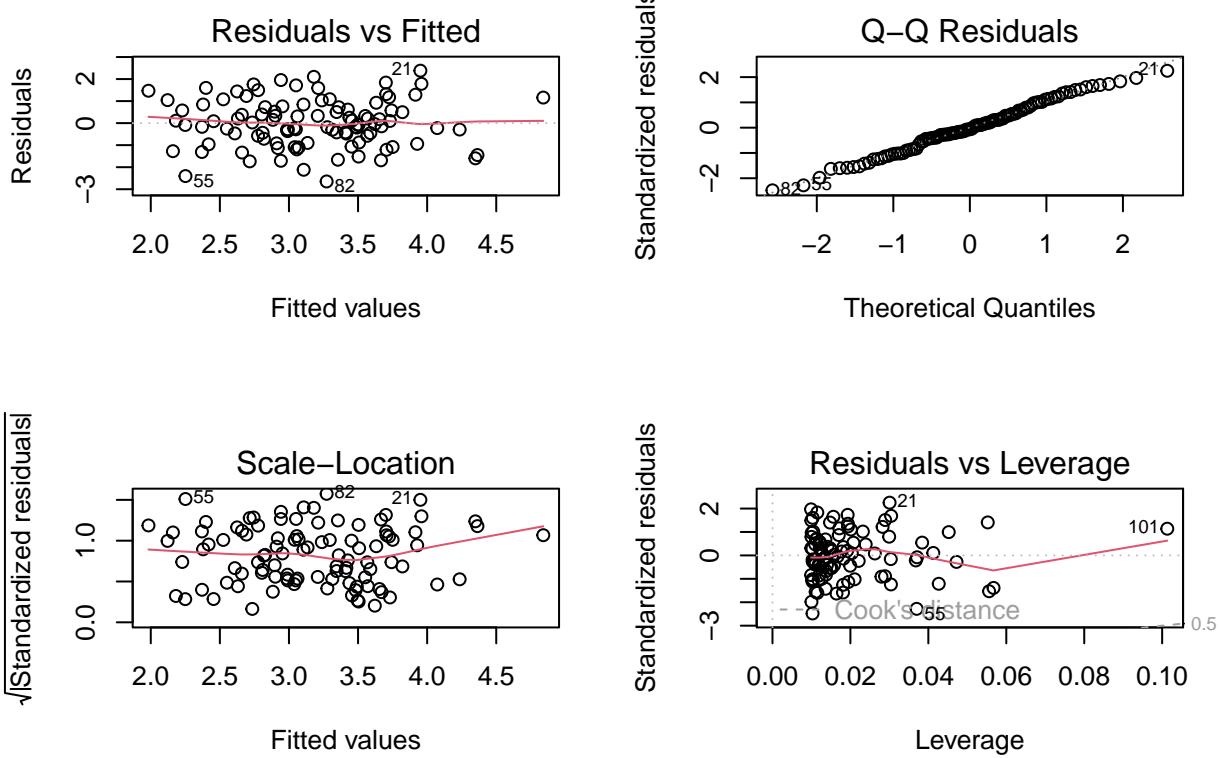
```
par(mfrow=c(2,2))
plot(fitg1)
```



```
par(mfrow=c(2,2))
plot(fitg2)
```



```
par(mfrow=c(2,2))
plot(fitg3)
```



We see that the first model now has  $X_2$  being statistically significant meaning we can reject the null hypothesis, while  $X_1$  is now not statistically significant and we fail to reject the null in its case. Both of the coefficient estimates are still inaccurate.

For the second and third models, we see that both  $X_1$  and  $X_2$  are both statistically significant for each of their respective regressions, meaning we can reject the null in both cases. For the model with  $X_1$ , we see that it still comes close to its true  $\beta$  estimate, but not as close as it did before this new observation was introduced. The  $\beta$  estimate for  $X_2$  is still way off for its respective model.

Looking at the residual plots for each model, we see that there is strong evidence across the board that the new observation is a high leverage point in the first model with both variables since it crosses the highest threshold for Cook's distance in the leverage plot.

The added observation does not show up as a high leverage point in either of the models with just  $X_1$  or  $X_2$ . However, it does appear to have relatively high leverage for the third model with just  $X_2$ , just not enough to be a concern based on Cook's distance.

For the second model with just  $X_1$ , we have evidence that the additional point could be an outlier from looking at the QQ plot and the residuals vs fitted plots. However, we don't appear to have any evidence that the added observation is an outlier in the first or third model with both variables and just  $X_2$ , respectively.