1(a)

Inflexible is best because simple models have enough data to learn the relationship with a large sample size. Also, flexible methods can overfit with few predictors.

1(b)

Flexible is best because it can capture complex relationships with higher dimensional data. Inflexible models are likely to underfit the data given the small amount of data.

1(c)

Flexible is best because it can capture non-linear relationships. Inflexible is too simple and will most likely underfit our data if it's non-linear.

1(d)

Inflexible is best because it generalizes noisy data with high variance. Flexible methods will likely overfit the data.

2(a)

i. False, provided that the GPA is not high, college graduates can generally earn more than high school graduates on average, as their base salary is \$35k higher.

ii. False, if the graduate's GPA is 4.0, our model predicts high school graduates will earn more than college graduates as the negative coefficient of the interaction effect between GPA and level outweighs the base salary boost of college graduates.

iii. True, when the GPA is high enough, the negative coefficient of the interaction effect between GPA and level outweighs the base salary boost for college graduates, resulting in high school graduates earning more on average.

iv. False, when the GPA is high enough, the negative coefficient of the interaction effect between GPA and level outweighs the base salary boost for college graduates, resulting in college graduates earning less on average.

2(b)

The model is as follows:

$$\hat{Y} = 50 + 20X_1 + 0.07X_2 + 35X_3 + 0.01X_1X_2 - 10X_1X_3$$

Given our inputs:

$$\hat{Y} = 50 + 20 * 4.0 + 0.07 * 110 + 35 + 0.01 * 4.0 * 110 - 10 * 4.0$$

$$\hat{Y} = 137.1$$

2(c)

False, although the coefficient for the interaction effect appears small, we cannot claim an interaction effect without conducting a statistical significance test.

3(a)

Generally, we would expect the training RSS of our cubic regression to appear lower than our linear regression, as it is more flexible and has more polynomial terms.

3(b)

Because of overfitting, we expect the testing RSS of our cubic regression to be higher than our linear regression. Our linear regression will have lower bias than the cubic regression.

3(c)

If the data is close to linear, we apply the reason from part (a), and we expect the training RSS of our cubic regression to appear lower than the linear regression. If the data is far from linear, then cubic regression is likely a better fit, resulting in a lower training RSS than our linear regression.

Note - The RSS formula is as follows:

$$RSS = \sum (Y_i - \hat{Y}_i)^2$$

We are assuming that the data follows a polynomial relationship for all parts.

3(d)

If the true relationship is close to linear, then apply the answer from part (b). However, if the true relationship is far from linear, then then we would expect test RSS of our cubic regression to be lower than our linear regression, as our linear regression likely underfitted the model. In this case, cubic regression wil have lower bias than linear regression.

Note - The RSS formula is as follows:

$$RSS = \sum (Y_i - \hat{Y}_i)^2$$

We are assuming that the data follows a polynomial relationship for all parts.

4

Let:

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$$

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

$$Cor(X,Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Since $\bar{x} = \bar{y} = 0$:

$$Cor(X,Y) = \frac{\sum_{i=1}^{n}(x_i y_i)}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i y_i)}{\sum_{i=1}^{n} x_i^2}$$

Solve for R^2:

$$R^2 = 1 - \frac{RSS}{TSS}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{\beta}_1 x_i)^2}{\sum_{i=1}^{n} y_i^2}$$

$$R^2 = \frac{\sum_{i=1}^{n}(2\hat{\beta}_1 x_i y_i - (\hat{\beta}_1 x_i)^2)}{\sum_{i=1}^{n} y_i^2}$$

$$R^2 = \frac{2\hat{\beta}_1 \sum_{i=1}^{n}(x_i y_i) - \hat{\beta}_1^2 \sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} y_i^2}$$

Plugging in $\hat{\beta}_1$:

$$R^2 = \frac{2\frac{\sum_{i=1}^{n}(x_i y_i)}{\sum_{i=1}^{n} x_i^2}\sum_{i=1}^{n}(x_i y_i) - \left(\frac{\sum_{i=1}^{n}(x_i y_i)}{\sum_{i=1}^{n} x_i^2}\right)^2 \sum_{i=1}^{n} x_i^2}{\sum_{i=1}^{n} y_i^2}$$

$$R^2 = \frac{2\frac{(\sum_{i=1}^{n}(x_i y_i))^2}{\sum_{i=1}^{n} x_i^2} - \frac{(\sum_{i=1}^{n}(x_i y_i))^2}{\sum_{i=1}^{n} x_i^2}}{\sum_{i=1}^{n} y_i^2}$$

$$R^2 = \frac{\frac{(\sum_{i=1}^{n}(x_i y_i))^2}{\sum_{i=1}^{n} x_i^2}}{\sum_{i=1}^{n} y_i^2}$$

$$R^2 = \frac{\sum_{i=1}^{n}(x_i y_i)^2}{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2}$$

$$R^2 = Cor(X,Y)^2$$

5(a)

```r
# Load required packages
library(ISLR)
library(ggplot2)

# Load the Auto dataset
data("Auto")
```

```r
# (a) Fit simple linear regression: mpg ~ horsepower
lm_fit <- lm(mpg ~ horsepower, data = Auto)
lm_summary <- summary(lm_fit)
lm_summary
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

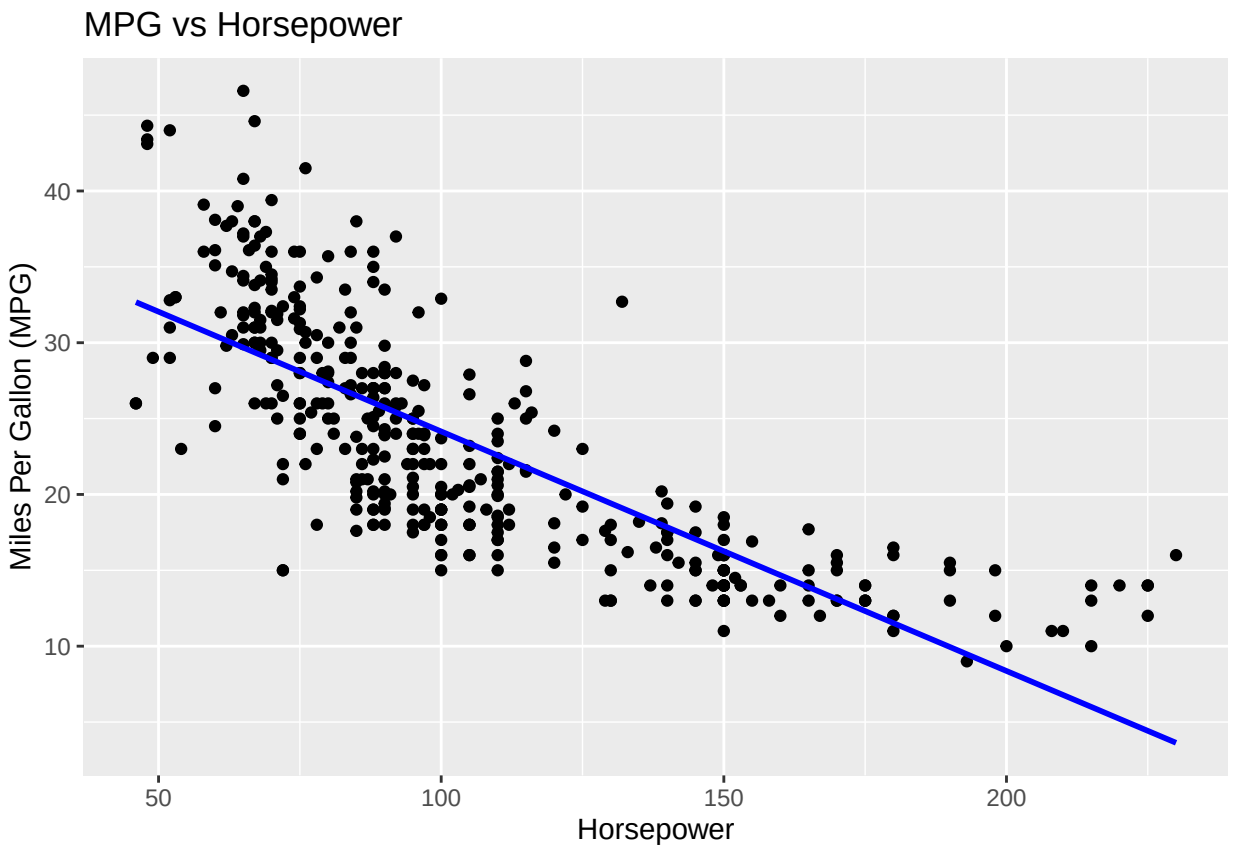
```r
# (a) iv. make prediction
test_data <- data.frame(horsepower = 98)
prediction <- predict(lm_fit, newdata = test_data)[1]

# (a) iv. get the 95% confidence and prediction interval
conf_interval <- predict(lm_fit, newdata = test_data, interval = "confidence")
pred_interval <- predict(lm_fit, newdata = test_data, interval = "prediction")
```

i. Yes, because $p << 0.001$, indicating there is a significant relationship between horsepower and mpg.
ii. The R^2 value is around 0.6059 which means that 60.59% of variability in mpg is explained by horsepower.
iii. The relationship is negative since the coefficient for horsepower is $-0.1578 < 0$.
iv. The prediction for 98hp is 24.4671mpg. The 95% confidence interval is $[23.9731, 24.9611]$. The 95% prediction interval is $[14.8094, 34.1248]$.

5(b)

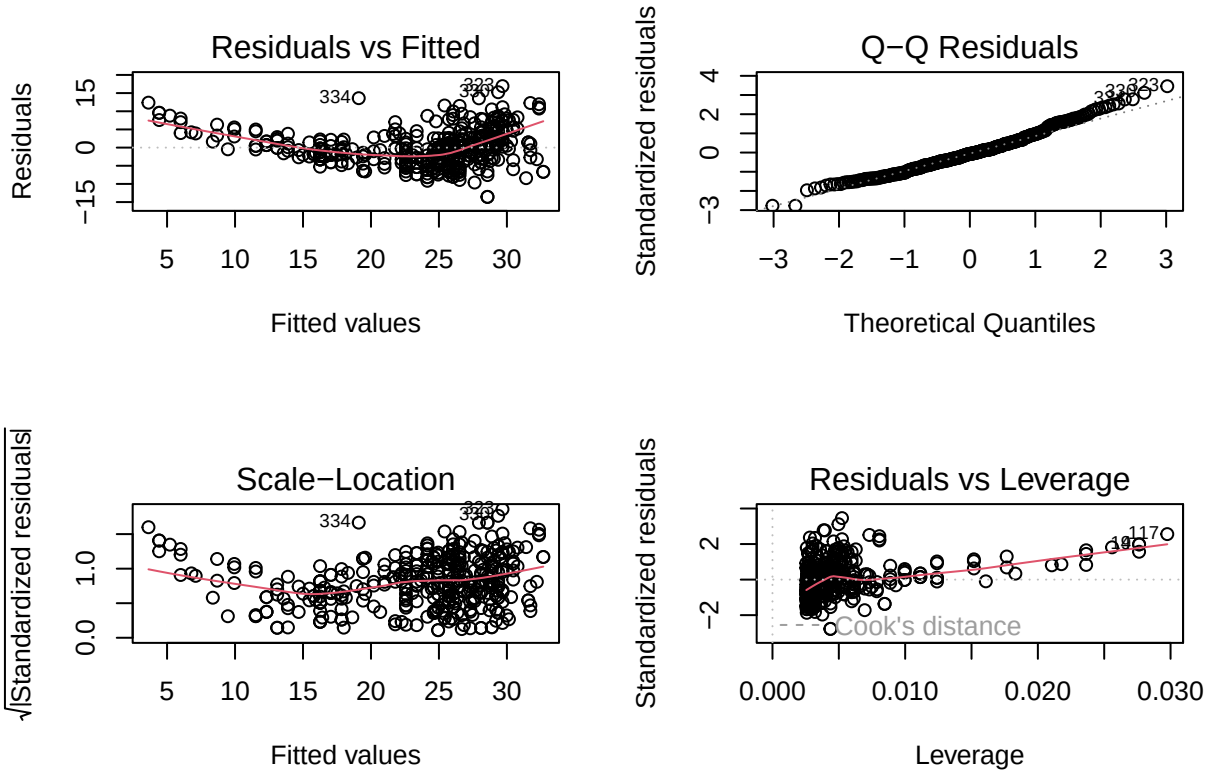
```
# (b) Scatterplot with regression line
ggplot(Auto, aes(x = horsepower, y = mpg)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "MPG vs Horsepower",
       x = "Horsepower",
       y = "Miles Per Gallon (MPG)")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



5(c)

```r
# (c) Diagnostic plots
par(mfrow = c(2, 2))  # Show all 4 plots in one window
plot(lm_fit)
```



i. The residuals vs fitted shows that the errors are not white noise. The data exhibits a pattern, showing that there may be an uncaptured underlying relationship. Utilizing a more flexible model or transforming the data may help.

ii. The data appears normal on the chart, as the points mostly corresopnd with the line. However, we cannot conclude normality without a formal test.

iii. The constant variance assumption is violated as there appears to be more residual variance for higher fitted values.

iv. It appears that some points above leverage 0.01 may be more influential and skewing the regression. Data points with |standardized residual| ≥ 2 may also skew the regression. Removing outliers may help.