Chapter 12: Statistics with R - 2nd Edition

Robert Stinerock

Student Exercises

The following exercises are intended to (1) provide practice analyzing data using simple linear regression and (2) review and reinforce our ability to subset data. The reason we emphasize these two skills together is that, in many instances, we want to analyze data that include only certain observations (and variables) while excluding the others. To this end, we make use of the `Cars93` data that is part of the `MASS` package. And because regression is the final analytic methodology of the book, we revisit several of the most useful functions covered earlier in the book, just for practice. The only csv data set used in these exercises—`polling.csv`—can be found on the website.

1. Import the `Cars93` data into the object named `E12_1`. What are the variable names? How many observations are included? Find (1) the minimum and maximum values, (2) the median and mean, (3) the first and third quartiles, and (4) the standard deviation of the two variables, `MPG.city` and `EngineSize`. Comment.

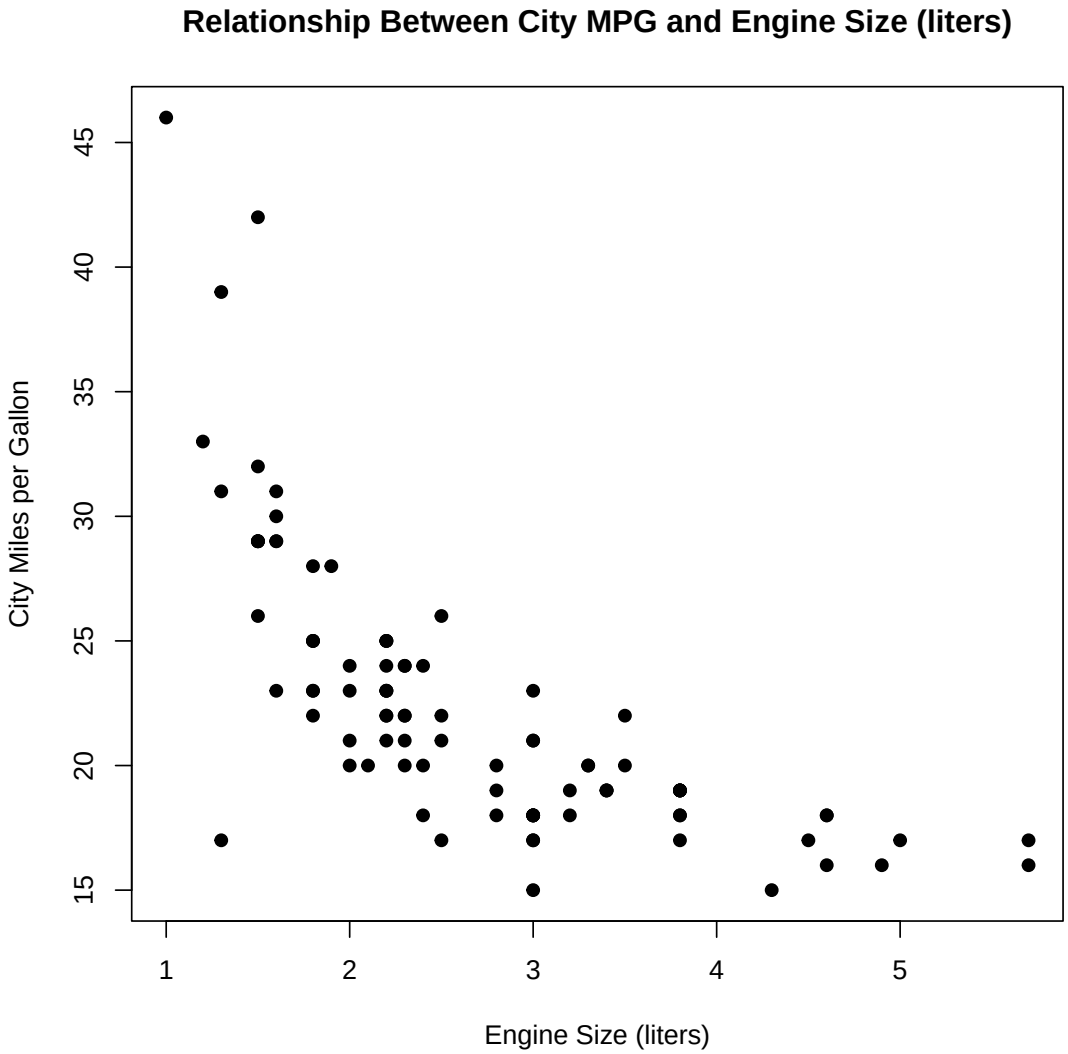
```
library(MASS) # Load the MASS package.



E12_1 <- Cars93 # Import Cars93 into the object named E12_1.



nrow(E12_1) # To find number of observations.

## [1] 93

names(E12_1) # To identify the variable names.

##  [1] "Manufacturer"       "Model"              "Type"
##  [4] "Min.Price"          "Price"              "Max.Price"
##  [7] "MPG.city"           "MPG.highway"        "AirBags"
## [10] "DriveTrain"         "Cylinders"          "EngineSize"
## [13] "Horsepower"         "RPM"                "Rev.per.mile"
## [16] "Man.trans.avail"    "Fuel.tank.capacity" "Passengers"
## [19] "Length"             "Wheelbase"          "Width"
## [22] "Turn.circle"        "Rear.seat.room"     "Luggage.room"
## [25] "Weight"             "Origin"             "Make"
```

```r
# Use the summary() function to find the basic descriptive
# statistics for MPG.city and EngineSize.

summary(E12_1$MPG.city)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.00   18.00   21.00   22.37   25.00   46.00

summary(E12_1$EngineSize)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.800   2.400   2.668   3.300   5.700

# Use the sd() function to find the standard deviation
# of each variable.

sd(E12_1$MPG.city)

## [1] 5.619812

sd(E12_1$EngineSize)

## [1] 1.037363
```

Answer: The `Cars93` data include 93 observations across 27 variables. The descriptive statistics for `MPG.city` and `EngineSize` are provided above.

2. Do `MPG.city` and `EngineSize` appear related in any systematic way? Comment.

```r
plot(E12_1$EngineSize, E12_1$MPG.city,
     pch = 19,
     xlab = 'Engine Size (liters)',
     ylab = 'City Miles per Gallon',
     main = 'Relationship Between City MPG and Engine Size (liters)')
```

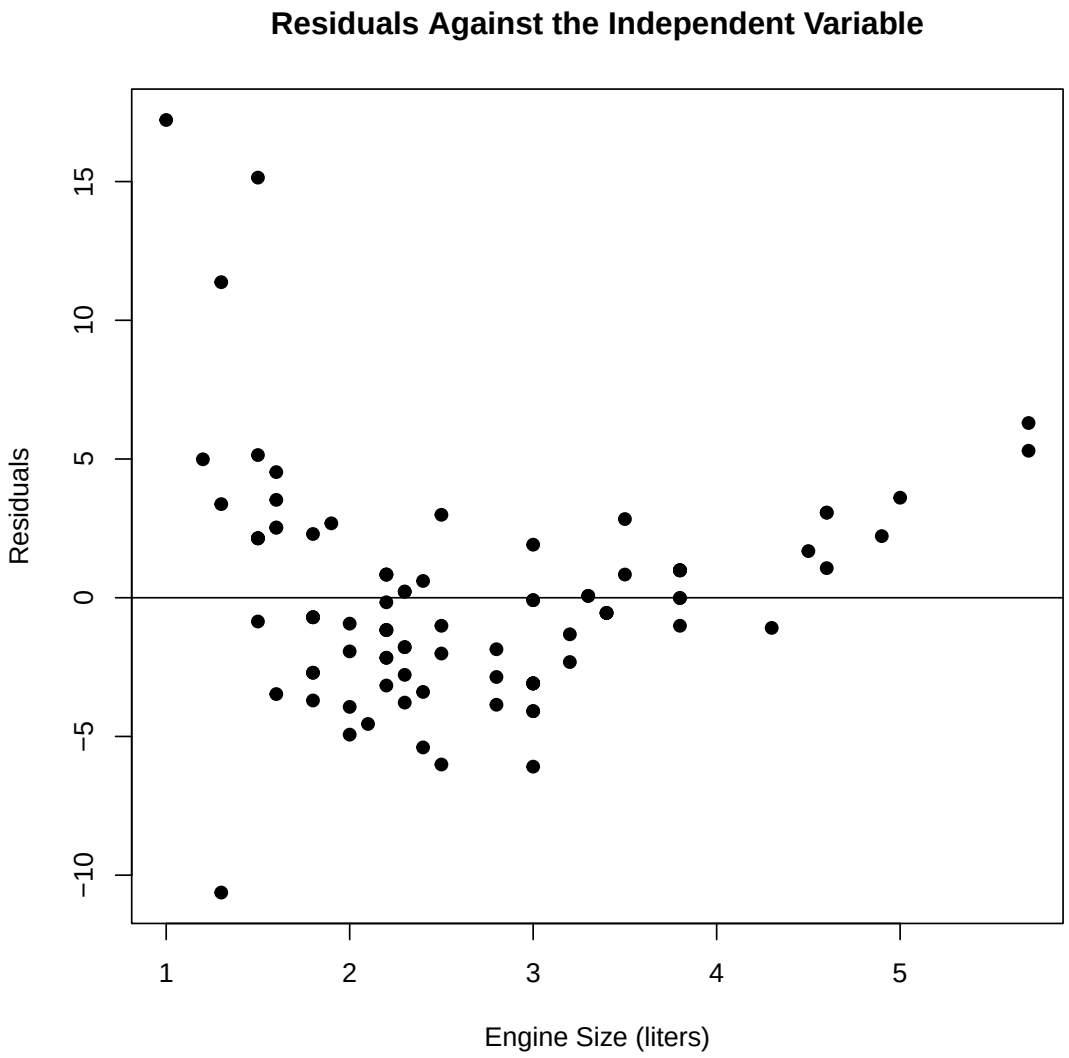


Answer: The pattern of points revealed by the scatterplot suggests that the relationship is negatively related. One important question is whether the relationship is linear; the scatterplot suggests that it is probably more curvilinear than linear. To sort out this issue, we need to consider the residual plot.

3. Make and inspect a residual plot. Does the pattern of points reveal anything that might cause us to question the assumptions underlying the appropriate usage of regression analysis to explore the relationship between `MPG.city` and `EngineSize`?

```
# Use the lm() function to create the model object;
# name it slr1 (the first simple linear regression).

slr1 <- lm(MPG.city ~ EngineSize, data = E12_1)

# Use the plot() function to create a residual plot.
# Note that resid(slr1) must be included as an argument.
```

```r
plot(E12_1$EngineSize,
     resid(slr1),
     abline(h = 0),
     pch = 19,
     xlab = 'Engine Size (liters)',
     ylab = 'Residuals',
     main = 'Residuals Against the Independent Variable')
```



Answer: When the engine size is between (roughly) 1.5 and 3.5 liters, the residuals reveal a reasonably linear relationship between `MPG.city` and `EngineSize`. However, this pattern tends to break down for both the upper and lower values of engine size: for vehicles having the smallest engine size (below 1.5 liters) and the largest engine size (above 3.5 liters) the pattern of the residuals tells us that the assumptions underlying the correct application of regression are not very well met.

4. There are several possible methods for managing the problem of nonlinear relationships among variables, such as what we have encountered in this case. One of the approaches involves transforming the variables—by way of logarithms, exponents,

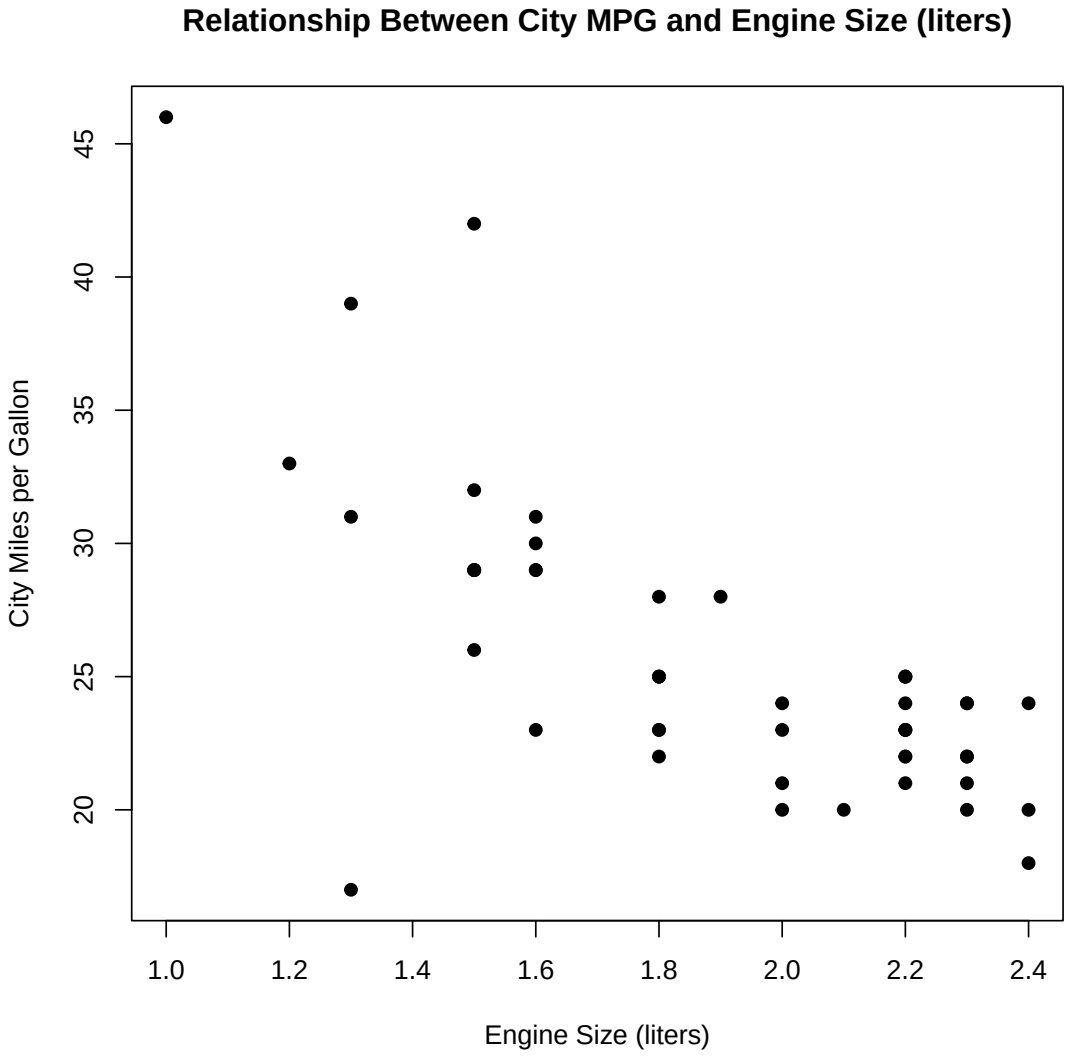
etc.—in such a way that they are *forced* to be more linearly related. (This class of methods, sometimes referred to as GLM or *general linear model*, is not covered in this book.) Another procedure requires including additional variables into the multiple regression model (the focus of Chapter 13). Instead, the approach we employ here involves subsetting the data according to some specification—such as, subsetting the data in a way that includes, for example, only vehicles manufactured in the US or all vehicles that have smaller engines (i.e., fewer liters of displacement). The expectation (or hope) is that, by subsetting, the resulting data may meet the assumptions behind the appropriate application of regression analysis. As a first step, subset the data stored in object `E12_1` in a way that excludes all vehicles with `EngineSize` greater than the median. Name this new object `E12_2`. Check to make sure that `E12_2` conforms to this requirement. How many observations remain in the new object? List the first three observations; list the last three observations.

```
# Use indexing [ , ]; set median Engine Size value of 2.4.

E12_2 <- E12_1[E12_1$EngineSize <= 2.40, c('MPG.city', 'EngineSize')]


# Use the max() and min() functions to find the maximum
# and minimum values of EngineSize in E12_2.

max(E12_2$EngineSize)

## [1] 2.4

min(E12_2$EngineSize)

## [1] 1

# List the first three and last three observations.

head(E12_2, 3)

##    MPG.city EngineSize
## 1        25        1.8
## 6        22        2.2
## 12       25        2.2

tail(E12_2, 3)

##    MPG.city EngineSize
## 90       21        2.0
## 92       21        2.3
## 93       20        2.4

# Use nrow() function to find the number of observations.

nrow(E12_2)

## [1] 49
```

Answer: The variable `EngineSize` now runs from a low of 1 to a high of 2.4 liters; `E12_2` now includes $n = 49$ observations.

5. For `E12_2`, do `MPG.city` and `EngineSize` appear related in a systematic way?

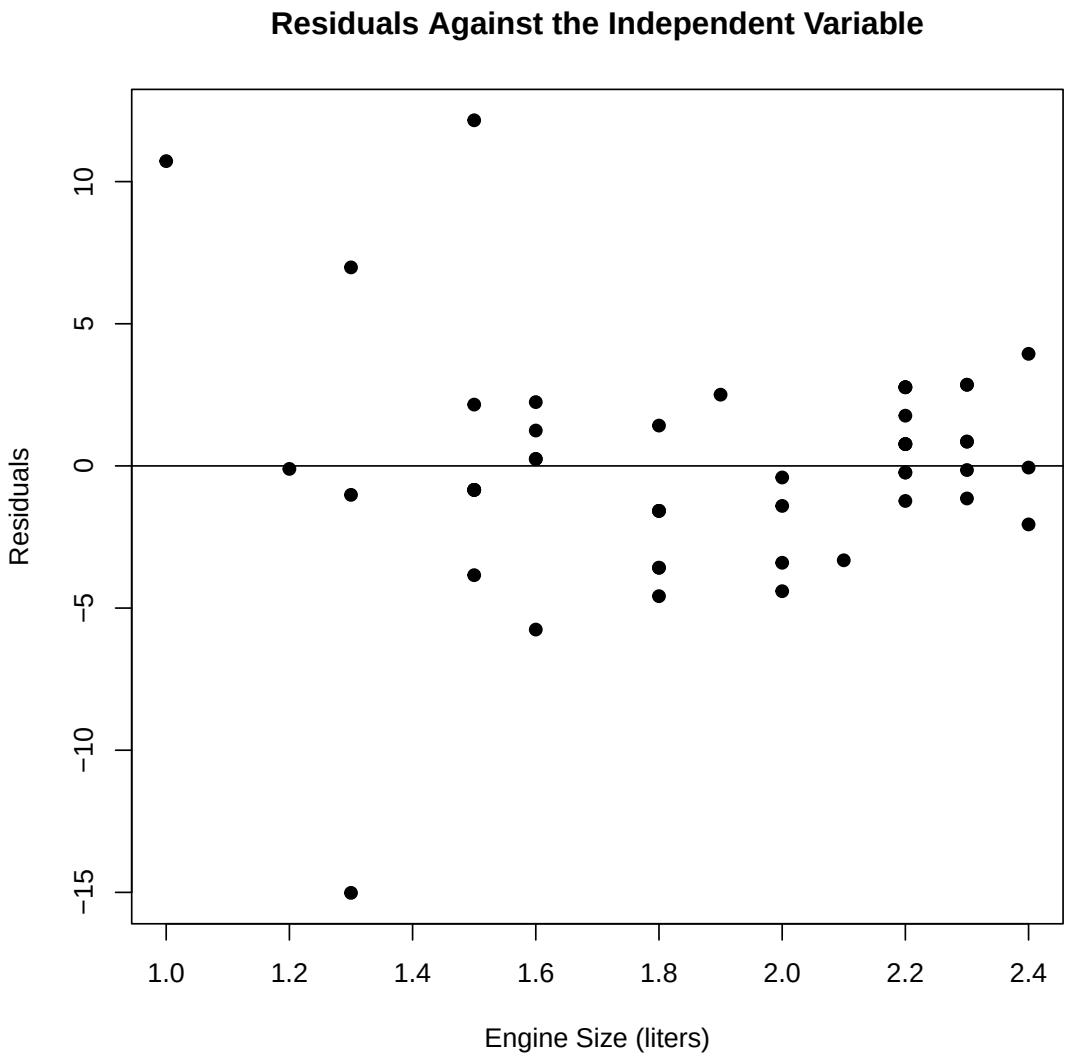
```
plot(E12_2$EngineSize, E12_2$MPG.city,
     pch = 19,
     xlab = 'Engine Size (liters)',
     ylab = 'City Miles per Gallon',
     main = 'Relationship Between City MPG and Engine Size (liters)')
```



Answer: Yes, the pattern of points revealed in the scatterplot of the `E12_2` data suggests that the two variables, `MPG.city` and `EngineSize`, may be negatively and linearly related. This is not a surprising finding, of course, since it confirms what we believe about this relationship in the first place.

6. Make and inspect a residual plot. Does the pattern of points appear more linearly related than they did before we subset the origin data?

```r
# Use the lm() function to create the model object named
# slr2 (the second simple linear regression).

slr2 <- lm(MPG.city ~ EngineSize, data = E12_2)


# Use the plot() function to create a residual plot.
# Note that resid(slr2) must be included as an argument.

plot(E12_2$EngineSize,
    resid(slr2),
    abline(h = 0),
    pch = 19,
    xlab = 'Engine Size (liters)',
    ylab = 'Residuals',
    main = 'Residuals Against the Independent Variable')
```



Answer: Yes, apart from three or four outliers for values of engine size below 1.5

liters, the residual plot does not reveal serious violations of the assumptions.

7. As part of making the residual plot in the preceding exercise, we used the `lm()` function to create the model object `slr2`. This is an important step in residual analysis because the model object (`slr2`) includes all the important information associated with the particular regression problem at hand, including the estimated regression equation itself. What is the estimated regression equation?

```
slr2


##
## Call:
## lm(formula = MPG.city ~ EngineSize, data = E12_2)
##
## Coefficients:
## (Intercept)   EngineSize
##       46.15       -10.87
```

Answer: The estimated regression equation is $\hat{y} = b_0 + b_1 x = 46.15 - 10.87x$, where $\hat{y}$ is the predicted dependent variable, `MPG.city`, and x is the independent variable, `EngineSize`.

8. Find the 95 and 99 percent confidence interval estimates of the regression coefficient b_1. Describe what these confidence intervals mean.

```
# Use the confint( , level = ) function to find the
# confidence interval estimates of the regression coefficient.

confint(slr2, level = 0.95)

##                 2.5 %    97.5 %
## (Intercept)  40.09671 52.207573
## EngineSize  -14.03045 -7.714777

confint(slr2, level = 0.99)

##                 0.5 %    99.5 %
## (Intercept)  38.07151 54.232777
## EngineSize  -15.08657 -6.658656
```

Answer: There is a 95% probability that the regression coefficient falls in the interval from -14.03045 to -7.714777; there is a 99% probability that it falls in the interval from -15.08657 to -6.658656.

9. What does the estimated regression equation tell us?

Answer: We can interpret the estimated regression equation $\hat{y} = 46.15 - 10.87x$ this way: for the class of vehicles with engine sizes of 2.4 liters or less, we expect that a change of 1 liter in engine size will be associated with a change of 10.87 miles per gallon when the vehicle is driving in a city. Moreover, the negative sign tells us that `MPG.city` and `EngineSize` are inversely related: as `EngineSize` increases (decreases), `MPG.city` decreases (increases). We know this because the regression coefficient $b_1 = -10.87$. The intercept term $b_0 = 46.15$ means less to us, except when we make predictions, because it implies that a vehicle with 0 liters of displacement should get 46.15 miles per gallon in city driving.

10. What is the strength of association between the two variables, `MPG.city` and `Engine Size`? Find the coefficient of determination r^2 using the following expression for r^2 (do not use the `summary()` function to unpack the regression statistics; we will use it later). This exercise provides another opportunity to hone your coding skills.

$$r^2 = \frac{\sum(y_i - \overline{y})^2 - \sum(y_i - \hat{y}_i)^2}{\sum(y_i - \overline{y})^2} = \frac{SS_y - SS_{res}}{SS_y}$$

```
# Find the total sum of squares, ss_y.

ss_y <- sum((E12_2$MPG.city - mean(E12_2$MPG.city)) ^ 2)


# Find the residual sum of squares, ss_res.

ss_res <- sum((resid(slr2)) ^ 2)


# Find the coefficient of determination.

(ss_y - ss_res) / ss_y

## [1] 0.505143
```

Answer: The coefficient of determination, $r^2 = 0.505143$.

11. What does the coefficient of determination r^2 reveal about the regression model?

Answer: The r^2 indicates the proportion of variation in the dependent variable `MPG.city` that is explained (or accounted for) by variation in the independent variable `EngineSize`. In this case, that proportion is 0.505143, or roughly 51%. Moreover, because $r^2 = 0.505143$, we also know that almost 49% of the variation in `MPG.city` remains unaccounted for, even once the association with `EngineSize` has been considered.

12. What is the t value of the coefficient b_1 on the independent variable `EngineSize`? Do not use the `summary()` function but rather write out the code.

    Answer: $t = -6.926538$

    Because finding the answer to this question requires a slightly more complicated bit of code, we break up the solution into several pieces.

    (a) The expression for the t value is found by taking the ratio of the coefficient itself to the standard error.

    $$t = \frac{b_1}{s_{b_1}}$$

    (b) Finding the denominator (i.e., the standard error s_{b_1}) of the above expression requires calculating another ratio

    $$s_{b_1} = \frac{s_{y|x}}{\sqrt{\sum(x_i - \overline{x})^2}} = \frac{4.061098}{2.587174} = 1.569704$$

    where the numerator of this ratio $s_{y|x}$ is

    $$s_{y|x} = \sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{n-2}} = 4.061098$$

```
s_xy <- sqrt(sum((resid(slr2) ^ 2)) / (nrow(E12_2) - 2))

s_xy

## [1] 4.061098
```

    and where the denominator of this ratio is

    $$\sqrt{\sum(x_i - \overline{x})^2} = 2.587174$$

```
ssx <- sqrt(sum((E12_2$EngineSize - mean(E12_2$EngineSize)) ^ 2))

ssx

## [1] 2.587174
```

    The ratio can now be found by dividing the first value (above) by the second. This value is s_{b_1}.

```
sb1 <- s_xy / ssx

sb1

## [1] 1.569704
```

(c) The numerator of the t statistic requires the regression coefficient b_1

$$b_1 = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sum(x_i - \overline{x})^2} = -10.87261$$

```
b1 <- sum((E12_2$EngineSize - mean(E12_2$EngineSize)) *

  (E12_2$MPG.city - mean(E12_2$MPG.city))) /

  sum((E12_2$EngineSize - mean(E12_2$EngineSize)) ^ 2)


b1

## [1] -10.87261
```

(d) Finally, the t statistic is found by dividing the regression coefficient b_1 by the standard error s_{b_1}.

$$t = \frac{b_1}{s_{b_1}} = \frac{-10.87261}{1.569704} = -6.926538$$

```
t = b1 / sb1

t

## [1] -6.926538
```

13. What is the p-value of $t = -6.926538$?

Answer: the p-value$=(2)(p(t \leq t = -6.926538, df = 47)) = 0.00000001056443$.

Note: For convenience and accuracy, we use t from the preceding exercise as the first argument of the `pt()` function.

```
# Since the p-value statistic has a very small value, we can
# override the default of reporting it in scientific notation.

options(scipen = 999)
```

```
# Use the pt() function with (n-2)=47 degrees of freedom.
# Remember that since this is a two-tail test, we need to multiply
# by 2.


2 * pt(t, 47)

## [1] 0.00000001056443
```

14. Use the `summary()` extractor function to check our work. Remember to use the model object `slr2` as the argument.

```
#Comment Use summary() function to extract the desired statistics.

summary(slr2)

##
## Call:
## lm(formula = MPG.city ~ EngineSize, data = E12_2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.0177  -1.5814  -0.1451   1.7676  12.1568
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept)    46.15       3.01  15.333 < 0.0000000000000002 ***
## EngineSize    -10.87       1.57  -6.927          0.0000000106 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.061 on 47 degrees of freedom
## Multiple R-squared:  0.5051,Adjusted R-squared:  0.4946
## F-statistic: 47.98 on 1 and 47 DF,  p-value: 0.00000001056
```

All the findings arrived at using the `summary()` function confirm what has been found in the preceding exercises. That is, the estimated regression equation is $\hat{y} = 46.15 - 10.87x$; the coefficient of determination is $r^2 = 0.505143$; the t statistic is $t = -6.926538$; and the p-value=0.00000001056443.

15. Use the regression equation to find the predicted values of `MPG.city` for the following values of `EngineSize` (liters of displacement): 1.25, 1.50, 1.75, 2.00, 2.25.

Answer: The predicted values of `MPG.city` for `EngineSize` of 1.25, 1.50, 1.75, 2.00, and 2.25 liters are (in order) 32.56138, 29.84322, 27.12507, 24.40692, and 21.68876 miles per gallon.

```
# Use data.frame() to create a new object containing 1.25,
# 1.50, 1.75, 2.00, and 2.25. Name the new object size_new.

size_new <- data.frame(EngineSize <- c(1.25, 1.50, 1.75, 2.00, 2.25))


# Use the predict() function to provide predicted values
# of miles per gallon for vehicles having 1.25, 1.50, 1.75, 2.00, and
# 2.25 liters EngineSize.

predict(slr2, size_new)

##        1        2        3        4        5
## 32.56138 29.84322 27.12507 24.40692 21.68876
```

16. What are the predicted values of `MPG.city` that were used to calibrate the estimated regression equation $\hat{y} = 46.15 - 10.87x$? Import those predicted values into an object named `mileage_predicted` and list the first and last three elements.

```
# Use the fitted(slr2) function to create the predicted
# values of the dependent variable. Import those values into
# the object named mileage_predicted.

mileage_predicted <- fitted(slr2)


# Use the head(,3) and tail(,3) functions to list the
# first and final three values of the predicted value.

head(mileage_predicted, 3)

##        1        6       12
## 26.58144 22.23239 22.23239

tail(mileage_predicted, 3)

##       90       92       93
## 24.40692 21.14513 20.05787
```

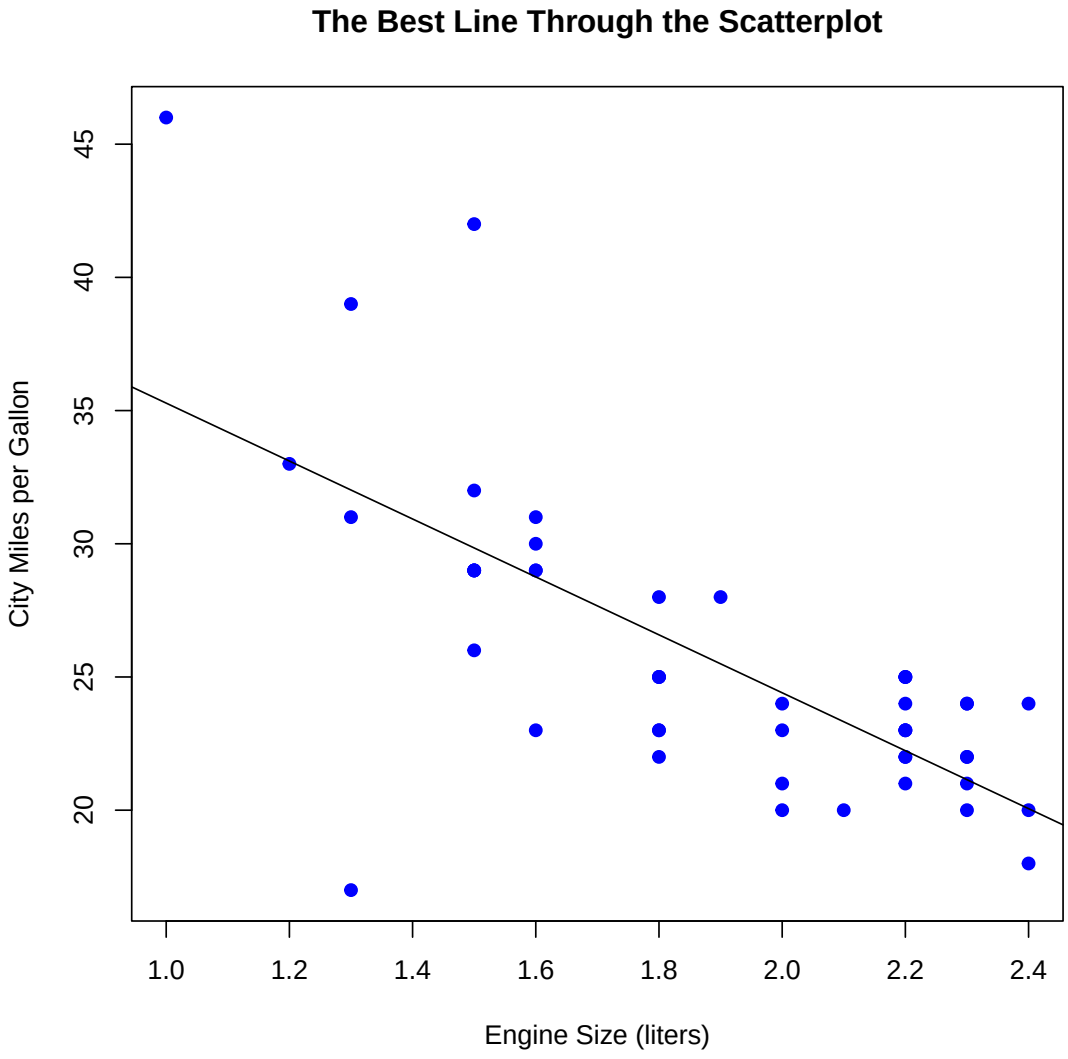
17. Add the `mileage_predicted` object (created in the preceding exercise) to `E12_2`, and name the resulting object `E12_3`. List the first and last four elements. Find the correlation of the actual and predicted variables; that is, the correlation of `MPG.city` and `mileage_predicted`. Once you have calculated the correlation, square it (i.e., raise it to the second power). Does the squared correlation coefficient look familiar?

```
# Use the cbind() function to bind the column
# mileage_predicted #to E12_2.  Name the new object E12_3.

E12_3 <- cbind(E12_2, mileage_predicted)


# List the first and last four elements of E12_3.

head(E12_3, 4)

##    MPG.city EngineSize mileage_predicted
## 1        25        1.8          26.58144
## 6        22        2.2          22.23239
## 12       25        2.2          22.23239
## 13       25        2.2          22.23239

tail(E12_3, 4)

##    MPG.city EngineSize mileage_predicted
## 88       25        1.8          26.58144
## 90       21        2.0          24.40692
## 92       21        2.3          21.14513
## 93       20        2.4          20.05787

# Find the correlation of the actual and predicted
# dependent variables. Store the value in an object named r.

r <- cor(E12_3$MPG.city, E12_3$mileage_predicted)

r

## [1] 0.7107341

# Square the value of r.

r^2

## [1] 0.505143
```

The square of the correlation of the *actual* dependent variable and *predicted* dependent variable equals the coefficient of determination, r^2.

18. Create a scatterplot with `MPG.city` on the vertical axis, `Engine Size` on the horizontal axis. Add labels to both axes as well as a main title. Finally, using the `abline()` function, add a regression line to the scatterplot.

```r
plot(E12_2$EngineSize, E12_2$MPG.city,
     xlab = 'Engine Size (liters)',
     ylab = 'City Miles per Gallon',
     main = 'The Best Line Through the Scatterplot',
     pch = 19,
     col = 'blue')

abline(slr2)
```



19. For additional practice structuring our data before analyzing it, we now subset `E12_1` by `Origin`. For this exercise: (1) create a new object from the original data set, `E12_1`, that includes only vehicles of non-USA origin (thus excluding all vehicles of USA origin) and name it `E12_4`; (2) find the median `EngineSize` of non-USA vehicles, and (3) create a new object, named `E12_5`, that includes only (a) those vehicles having `MPG.city` less than or equal to the median and only (b) the two variables, `MPG.city` and `EngineSize`. In other words, subset the original data set, `E12_1`, to include only the two variables, `MPG.city` and `EngineSize`, and only those

vehicles that are of non-USA origin and that feature engines with displacement (in liters) at or below the median for the relevant category. Just to make sure `E12_5` "looks" as it should, run a few of the same functions that were used in Exercise 1.

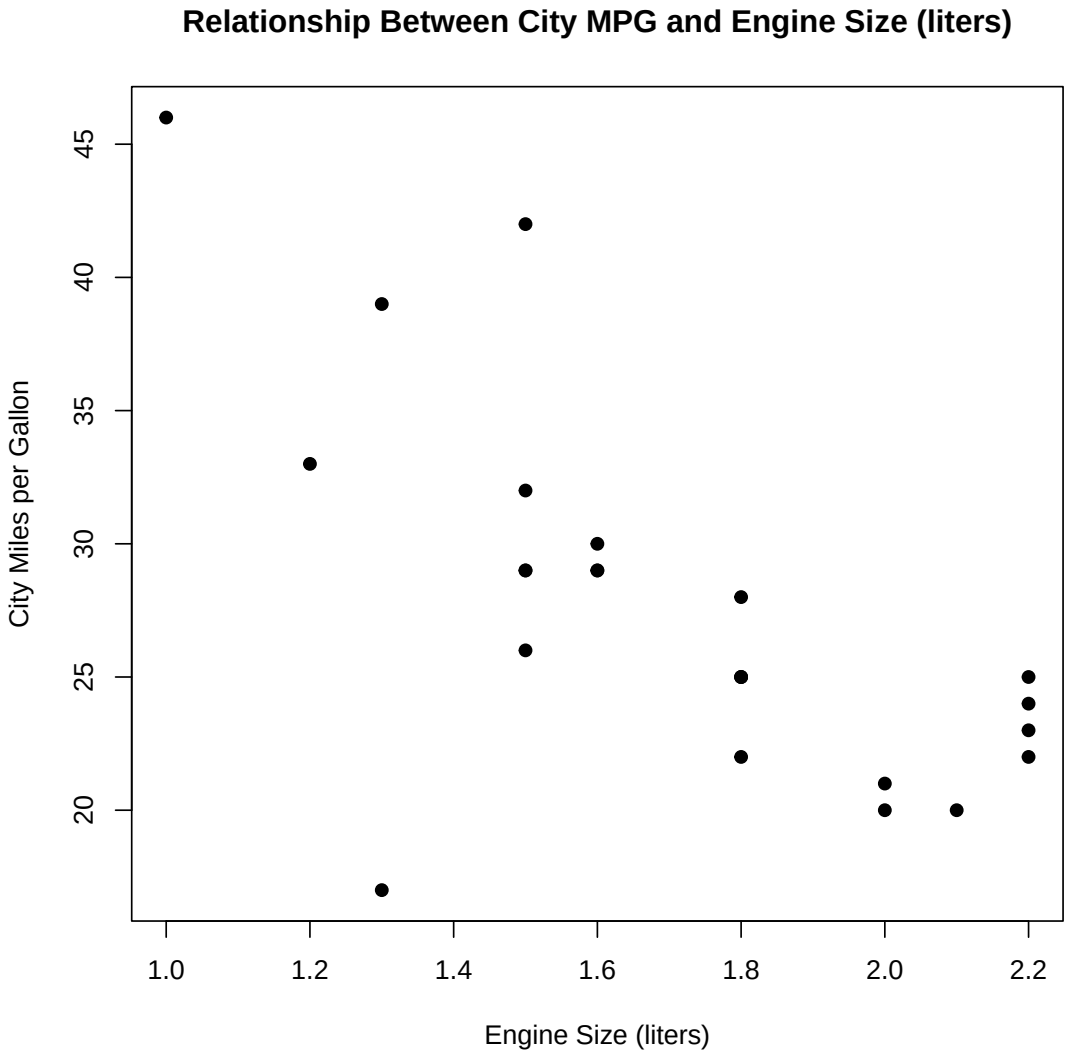
```
# Subset data so that it includes only vehicles of non-USA
# origin. Name new object E12_4.

E12_4 <- E12_1[ which(E12_1$Origin == 'non-USA'), ]


# Find the median EngineSize for the sample including only
# non-USA vehicles.

median(E12_4$EngineSize)

## [1] 2.2

# Since the median is 2.2 liters, subset the data once
# again to include only vehicles with 2.2 liters (or less) engine
# displacement. Name the new object E12_5.

E12_5 <- E12_4[E12_4$EngineSize <= 2.20, c('MPG.city', 'EngineSize')]


# Use the summary() function to find the basic descriptive
# statistics for MPG.city and EngineSize

summary(E12_5$MPG.city)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   22.75   25.50   27.54   29.25   46.00

summary(E12_5$EngineSize)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.500   1.700   1.708   2.000   2.200

# Use nrow() function to find the number of observations.

nrow(E12_5)

## [1] 24

# Use the names() function to confirm that E12_5 includes
#only two variables, MPG.city and EngineSize.

names(E12_5)
```

```
## [1] "MPG.city"   "EngineSize"

# Finally, print out the entire new (twice subsetted) data set.

E12_5

##    MPG.city EngineSize
## 1        25        1.8
## 39       46        1.0
## 40       30        1.6
## 42       42        1.5
## 43       24        2.2
## 44       29        1.5
## 45       22        1.8
## 46       26        1.5
## 47       20        2.0
## 53       29        1.6
## 54       28        1.8
## 57       17        1.3
## 62       29        1.5
## 64       29        1.6
## 78       20        2.1
## 80       33        1.2
## 81       25        1.8
## 82       23        2.2
## 83       39        1.3
## 84       32        1.5
## 85       25        2.2
## 86       22        2.2
## 88       25        1.8
## 90       21        2.0
```

Answer: The new data set, `E12_5`, includes 24 observations across the two variables, `MPG.city` and `EngineSize`. The descriptive statistics for both variables are reported above. Note that the maximum value of `EngineSize` is 2.2, thus confirming that the data include only what we want, in terms of observations, variables, and restrictions. This practice of "looking under the hood (or bonnet)" is a sound one. It allows us to confirm that the data really do look like they should.

20. For the category of vehicles of non-USA origin, do the two variables, `MPG.city` and `EngineSize`, seem to be related in a systematic way? If so, how?

```
plot(E12_5$EngineSize, E12_5$MPG.city,
     pch = 19,
     xlab = 'Engine Size (liters)',
     ylab = 'City Miles per Gallon',
     main = 'Relationship Between City MPG and Engine Size (liters)')
```

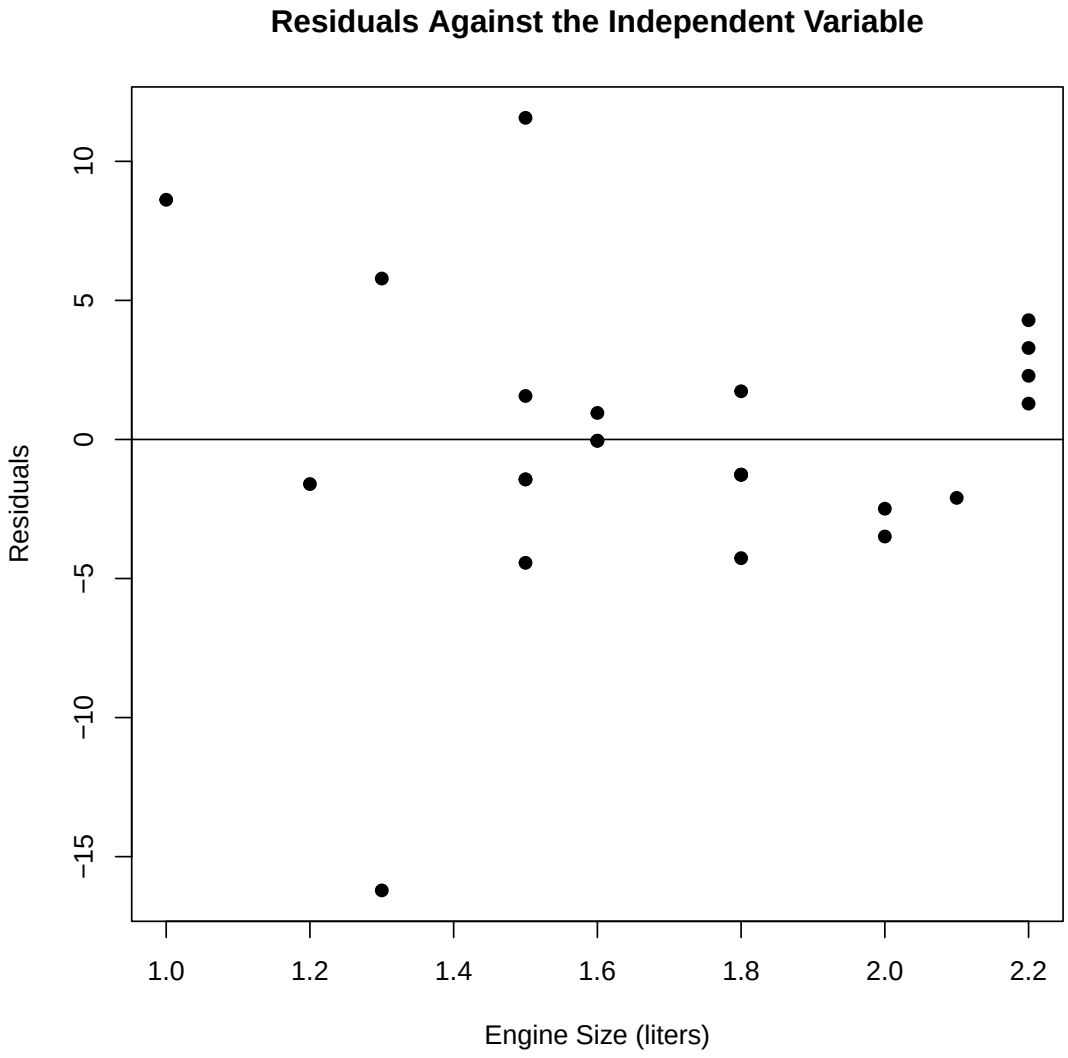


Answer: Yes, the pattern of points appears to run (approximately) from the upper-lefthand to lower-righthand corners of the scatterplot, implying a negative association: larger engine sizes are associated with lower miles per gallon (in city driving). But because of a few outlier cases, we do not expect the r^2 to be very high, perhaps not even $r^2 = 0.50$.

21. Make and inspect a residual plot. Does the pattern of points reveal any reason why we should not use regression to analyze these data? Are there any radical departures from the assumptions underlying the appropriate usage of this methodology?

```
# Use the lm() function to create the model object named
# slr3 (slr3 stands for the third simple linear regression).

slr3 <- lm(MPG.city ~ EngineSize, data = E12_5)


# Use the plot() function to create a residual plot.
# Note that resid(slr3) must be included as an argument.
```

```
plot(E12_5$EngineSize,
     resid(slr3),
     abline(h = 0),
     pch = 19,
     xlab = 'Engine Size (liters)',
     ylab = 'Residuals',
     main = 'Residuals Against the Independent Variable')
```



Answer: The only possible area of trouble revealed by the residual plot might be in the range of the smaller engines, particulary at and below 1.5 liters. For vehicles with larger engines, however, the assumption of constant variation seems satisfied. Therefore, although the data are far from what we might characterize as "well behaved," the violations do not seem serious enough to cause us to drop regression as a potentially promising analytic methodology.

22. What is the estimated regression equation?

```
slr3
```

```
##
## Call:
## lm(formula = MPG.city ~ EngineSize, data = E12_5)
##
## Coefficients:
## (Intercept)   EngineSize
##       51.27       -13.89
```

Answer: The estimated regression equation is $\hat{y} = 51.27 - 13.89x$. The predicted dependent variable is $\hat{y}$, `MPG.city`; the independent variable is x, or `EngineSize`. Note that to recover this minimal information, we need only enter the model object, `slr3`.

23. Find the 75 and 90 percent confidence interval estimates of the regression coefficient b_1. How should we interpret the meaning of these confidence interval estimates?

```
confint(slr3, level = 0.75)
```

```
##                12.5 %    87.5 %
## (Intercept)  44.60803  57.94082
## EngineSize  -17.72209 -10.06260
```

```
confint(slr3, level = 0.90)
```

```
##                   5 %      95 %
## (Intercept)  41.58615 60.962698
## EngineSize  -19.45811 -8.326578
```

Answer: There is a 75% probability that the regression coefficient falls in the interval from -17.72209 to -10.06260; there is a 90% probability that it falls in the interval from -19.45811 to -8.326578.

24. What does the estimated regression equation tell us?

Answer: The estimated regression equation $\hat{y} = 51.27 - 13.89x$ can be interpreted in this manner: for the category of vehicles of non-USA origin—and with engine sizes of 2.2 or fewer liters—we can expect that a change of 1 liter in engine size will be associated with a change of 13.89 miles per gallon when the vehicle is driving in a city. The negative sign, $b_1 = -13.89$, also tells us that as `EngineSize` increases (decreases), `MPG.city` decreases (increases). Although the intercept term, $b_0 = 51.27$, is not meaningfully interpretable, we will find it necessary to include it in the regression equation whenever we want to forecast or make predictions.

25. What is the strength of association between the two variables, `MPG.city` and `Engine Size`? Find the coefficient of determination r^2 using the following expression for r^2 (do not use the `summary()` function to unpack the regression statistics; we will use it later). This exercise provides another opportunity to hone your coding skills.

$$r^2 = \frac{\sum(y_i - \overline{y})^2 - \sum(y_i - \hat{y}_i)^2}{\sum(y_i - \overline{y})^2} = \frac{SS_y - SS_{res}}{SS_y}$$

```
# Find the total sum of squares, ss_y.

ss_y <- sum((E12_5$MPG.city - mean(E12_5$MPG.city)) ^ 2)


# Find the residual sum of squares, ss_res.

ss_res <- sum((resid(slr3)) ^ 2)


# Find the coefficient of determination.

(ss_y - ss_res) / ss_y

## [1] 0.455044
```

Answer: The coefficient of determination, $r^2 = 0.455044$.

26. What does the coefficient of determination r^2 tell us about the regression model?

Answer: The r^2 is the proportion of variation in the dependent variable `MPG.city` that is accounted for (or explained) by variation in `EngineSize`, the independent variable. When $r^2 = 0.455044$, we understand that that proportion is about 45%. We also know that approximately 55% of variation in `MPG.city` remains unexplained, even after taking `EngineSize` into account.

27. What is the t value of the coefficient b_1 on the independent variable `EngineSize`? Do not use the `summary()` function but rather write out the code (more practice).

Answer: $t = -4.286$

Because finding the answer to this question requires a slightly more complicated bit of code, we break up the solution into several pieces.

(a) The expression for the t value is found by taking the ratio of the coefficient itself to the standard error.

$$t = \frac{b_1}{s_{b_1}}$$

(b) Finding the denominator (i.e., the standard error s_{b_1}) of the above expression requires calculating another ratio

$$s_{b_1} = \frac{s_{y|x}}{\sqrt{\sum(x_i - \overline{x})^2}} = \frac{5.304575}{1.636561} = 3.241293$$

where the numerator of this ratio $s_{y|x}$ is

$$s_{y|x} = \sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{n-2}} = 5.304575$$

```
s_xy <- sqrt(sum((resid(slr3) ^ 2)) / (nrow(E12_5) - 2))

s_xy

## [1] 5.304575
```

and where the denominator of this ratio is

$$\sqrt{\sum(x_i - \overline{x})^2} = 1.636561$$

```
ssx <- sqrt(sum((E12_5$EngineSize - mean(E12_5$EngineSize)) ^ 2))

ssx

## [1] 1.636561
```

The ratio can now be found by dividing the first value (above) by the second. This is the value for s_{b_1}

```
sb1 <- s_xy / ssx

sb1

## [1] 3.241293
```

(c) The numerator of the t statistic requires the regression coefficient b_1

$$b_1 = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sum(x_i - \overline{x})^2} = -13.89235$$

```
b1 <- sum((E12_5$EngineSize - mean(E12_5$EngineSize)) *

  (E12_5$MPG.city - mean(E12_5$MPG.city))) /

  sum((E12_5$EngineSize - mean(E12_5$EngineSize)) ^ 2)

b1

## [1] -13.89235
```

(d) Finally, the t statistic is found by dividing the regression coefficient b_1 by the standard error s_{b_1}.

$$t = \frac{b_1}{s_{b_1}} = \frac{-13.89235}{3.241293} = -4.286051$$

```
t = b1 / sb1

t

## [1] -4.286051
```

28. What is the p-value of $t = -4.286051$?

Answer: p-value$= (2)(p(t \leq t = -4.286051, df = 22)) = 0.0003000032$.

Note: For convenience and accuracy, we use the t from the preceding exercise as the first argument of the `pt()` function.

```
# Use the pt() function with (n-2)=22 degrees of freedom.
# Remember that since this is a two-tail, we need to multiply by 2.

2 * pt(t, 22)

## [1] 0.0003000032
```

29. Use the `summary()` extractor function to check our work. Remember to use the model object `slr3` as the argument.

```
summary(slr3)


##
## Call:
## lm(formula = MPG.city ~ EngineSize, data = E12_5)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.2144  -1.7278  -0.6574   1.8710  11.5641
##
## Coefficients:
##             Estimate Std. Error t value      Pr(>|t|)
## (Intercept)   51.274      5.642   9.088 0.00000000668 ***
## EngineSize   -13.892      3.241  -4.286        0.0003 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.305 on 22 degrees of freedom
## Multiple R-squared:  0.455,Adjusted R-squared:  0.4303
## F-statistic: 18.37 on 1 and 22 DF,  p-value: 0.0003
```

All findings arrived at using the `summary()` function confirm what has been found in the preceding exercises. The estimated regression equation is $\hat{y} = 51.274 - 13.892x$; the coefficient of variation is $r^2 = 0.455$; the t statistic is $t = -4.286051$; and the p-value=0.0003000032.

30. Use the estimated regression equation to find the predicted values of `MPG.city` for the following values of `EngineSize` (liters of displacement): 1.25, 1.50, 1.75, 2.00, 2.25.

    Answer: The predicted values of `MPG.city` for `EngineSize` of 1.25, 1.50, 1.75, 2.00, and 2.25 liters are (in order) 33.90899, 30.43591, 26.96282, 23.48973, and 20.01665 miles per gallon.

```
# Use data.frame() to create a new object containing 1.25,
# 1.50, 1.75, 2.00, and 2.25. Name the new object size_new.

size_new <- data.frame(EngineSize <- c(1.25, 1.50, 1.75, 2.00, 2.25))


# Use predict() function to provide the predicted values
# of miles per gallon for vehicles having 1.25, 1.50, 1.75, 2.00, and
# 2.25 liters EngineSize.

predict(slr3, size_new)

##        1        2        3        4        5
## 33.90899 30.43591 26.96282 23.48973 20.01665
```

31. What are the predicted values of `MPG.city` that were used to calibrate the estimated regression equation $\hat{y} = 51.274 - 13.892x$? Import those predicted values into an object named `mileage_predicted` and list the first and last three elements.

```r
# Use fitted(slr3) function to create the predicted
# values of the dependent variable. Import those values into
# the object named mileage_predicted.

mileage_predicted <- fitted(slr3)


# Use the head(,3) and tail(,3) functions to list the
# first and final three values of the predicted value.

head(mileage_predicted, 3)
```

```
##        1       39       40
## 26.26820 37.38208 29.04667
```

```r
tail(mileage_predicted, 3)
```

```
##       86       88       90
## 20.71126 26.26820 23.48973
```

32. Merge the `mileage_predicted` object (created in the preceding exercise) with `E12_5`, and name the resulting object `E12_6`. List the first and last four elements. Find the correlation of the actual and predicted variables; that is, the correlation of `MPG.city` and `mileage_predicted`. Once you have the correlation, square it (i.e., raise it to the second power). Comment on the square of the correlation. What is it?

```r
# Use the cbind() function to bind the column
# mileage_predicted #to E12_5.  Name the new object E12_6.

E12_6 <- cbind(E12_5, mileage_predicted)


# List the first and last four elements of E12_6.

head(E12_6, 4)
```

```
##    MPG.city EngineSize mileage_predicted
## 1        25        1.8          26.26820
## 39       46        1.0          37.38208
## 40       30        1.6          29.04667
## 42       42        1.5          30.43591
```

```r
tail(E12_6, 4)
```

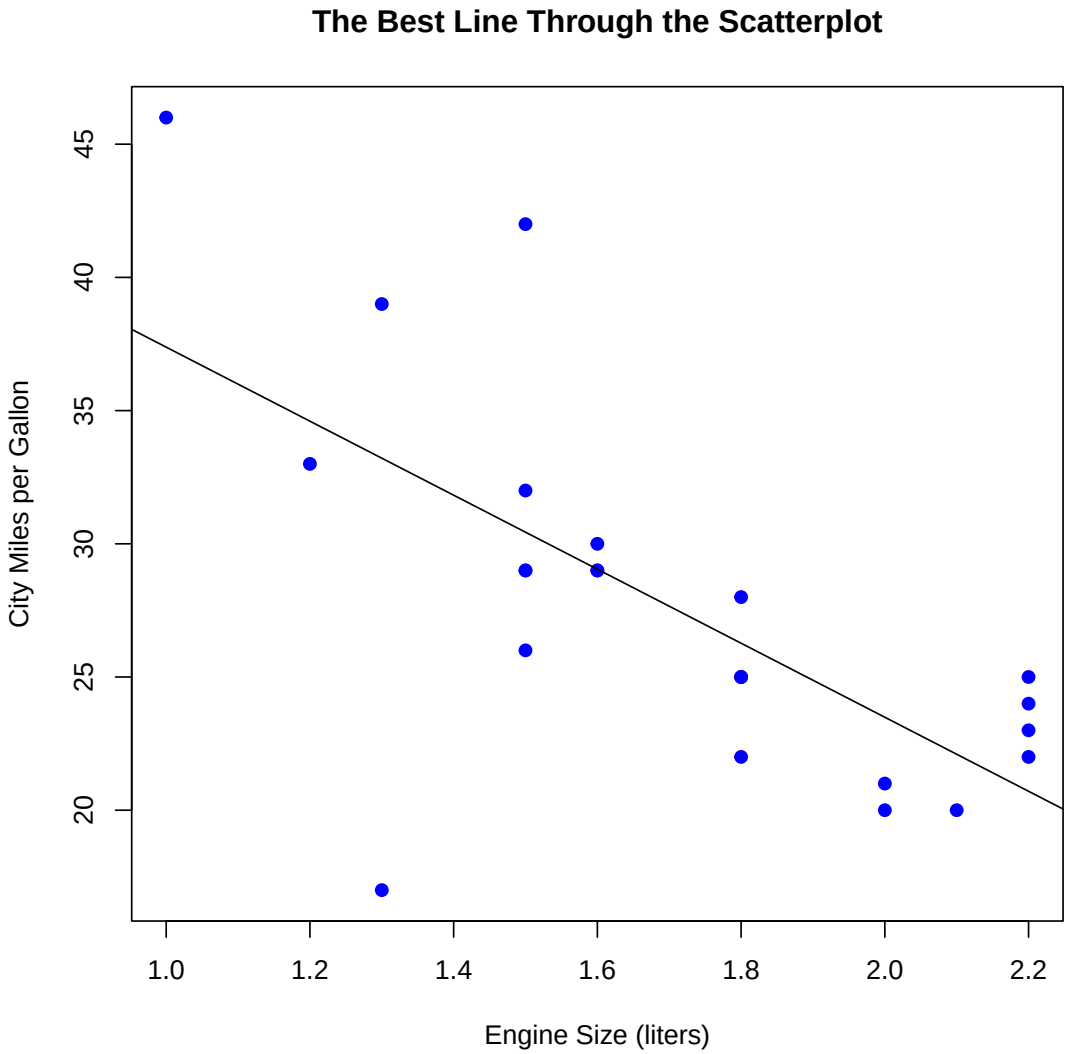
```
##    MPG.city EngineSize mileage_predicted
## 85       25        2.2          20.71126
```

```
## 86       22        2.2           20.71126
## 88       25        1.8           26.26820
## 90       21        2.0           23.48973

# Find the correlation of the actual and predicted
# dependent variables. Store the value in an object named r.

r <- cor(E12_6$MPG.city, E12_6$mileage_predicted)

r

## [1] 0.6745695

# Square the value of r.

r^2

## [1] 0.455044
```

The square of the correlation of the *actual* dependent variable and *predicted* dependent variable equals the coefficient of determination, r^2.

33. Create a scatterplot with `MPG.city` on the vertical axis, `Engine Size` on the horizontal axis. Add labels to both axes as well as a main title; set blue as the color of the points. Finally, using the `abline()` function, add a regression line to the scatterplot.

```
plot(E12_6$EngineSize, E12_6$MPG.city,
     xlab = 'Engine Size (liters)',
     ylab = 'City Miles per Gallon',
     main = 'The Best Line Through the Scatterplot',
     pch = 19,
     col = 'blue')

abline(slr3)
```



34. This exercise provides further opportunity to find data from an online source, create a data frame from scratch, and analyze it using some of the methods associated with simple linear regression. In this instance, we look up and record the high and low intra-day temperatures (in either degrees Celsius or Fahrenheit) for the following 14 cities from around the world: Auckland, Beijing, Cairo, Lagos, London, Mexico City, Mumbai, Paris, Rio de Janeiro, Sydney, Tokyo, Toronto, Vancouver, and Zurich.

    (a) Use the `c()` function to create three objects, one for each city name, one for the high temperature, and one for the low temperature. The data are recorded for a recent December day.

```
city <- c('Auckland', 'Beijing', 'Cairo', 'Lagos', 'London',
          'Mexico City', 'Mumbai', 'Paris', 'Rio de Janeiro',
          'Sydney', 'Tokyo', 'Toronto', 'Vancouver', 'Zurich')

high <- c(71, 45, 65, 91, 46, 67, 88, 44, 92, 88, 57, 20, 42, 40)
```

```
low <- c(56, 23, 48, 76, 37, 45, 71, 35, 73, 65, 39, 15, 39, 29)
```

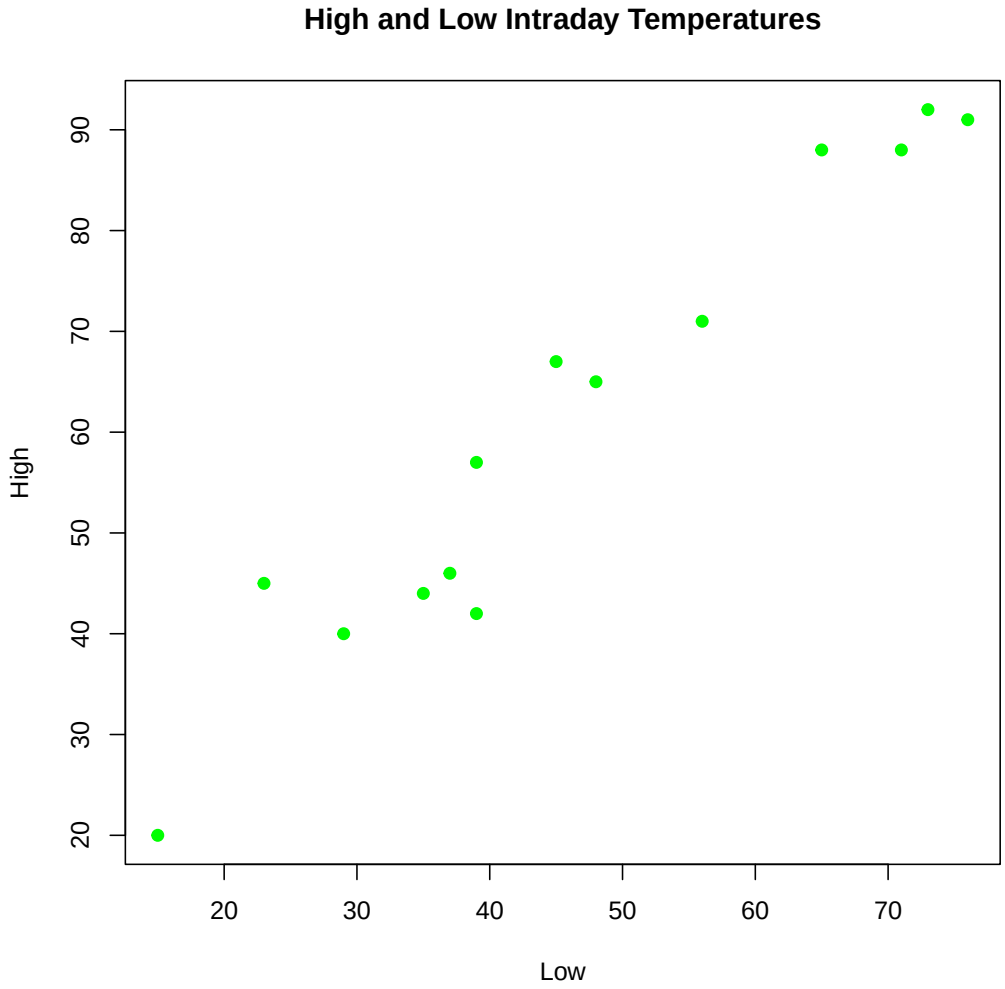
(b) Use the `data.frame()` function to create a data frame consisting of each city name and high and low temperatures. Display the results to check your work.

```
WorldTemps <- data.frame(City = city, High = high, Low = low)

WorldTemps
```

```
##              City High Low
## 1        Auckland   71  56
## 2         Beijing   45  23
## 3           Cairo   65  48
## 4           Lagos   91  76
## 5          London   46  37
## 6     Mexico City   67  45
## 7          Mumbai   88  71
## 8           Paris   44  35
## 9  Rio de Janeiro   92  73
## 10         Sydney   88  65
## 11          Tokyo   57  39
## 12        Toronto   20  15
## 13      Vancouver   42  39
## 14         Zurich   40  29
```

(c) Make a scatterplot of high against low temperatures. Create a main title, label each axis appropriately, and use `pch=` to specify how the points should appear. Does the pattern of points appear to confirm that the relationship between high and low temperatures is linear?

```
plot(WorldTemps$Low, WorldTemps$High,
     pch = 19,
     col = 'green',
     xlab = "Low",
     ylab = "High",
     main = "High and Low Intraday Temperatures")
```



Answer: The scatterplot makes clear that the relationship between high and low intra-day temperatures is both positive and linear.

(d) Estimate and write out the regression equation $\hat{y} = b_0 + b_1 x$. Let the high temperature be the dependent variable, and the low temp, the independent variable.

```
reg_eq_temps <- lm(High ~ Low, data = WorldTemps)

reg_eq_temps

##
## Call:
## lm(formula = High ~ Low, data = WorldTemps)
##
## Coefficients:
## (Intercept)          Low
##       7.763        1.148
```

The estimated regression equation is: $\hat{y} = 7.763 + 1.148x$

(e) What is the r^2?

```
summary(reg_eq_temps)

##
## Call:
## lm(formula = High ~ Low, data = WorldTemps)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.533  -3.991  -1.051   3.884  10.834
##
## Coefficients:
##             Estimate Std. Error t value     Pr(>|t|)
## (Intercept)  7.76313    4.27689   1.815       0.0946 .
## Low          1.14795    0.08546  13.433 0.0000000136 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.918 on 12 degrees of freedom
## Multiple R-squared:  0.9376,Adjusted R-squared:  0.9325
## F-statistic: 180.5 on 1 and 12 DF,  p-value: 0.00000001363
```

The r^2 is 0.9376, indicating that approximately 93.76% of variation in the dependent variable is explained by variation in the independent variable.

(f) What is the p-value? Is the estimate regression equation significant? Why or why not?

Since p-value=0.0000000136 is far less than the usual values we set for α—e.g., 0.05, 0.01, etc.—we say that the estimated regression equation is significant.

35. A dependent variable y is regressed on an independent variable x; the sample size is $n = 32$.

(a) If $SS_{reg} = 808.89$ and $SS_{res} = 317.16$, what is the r^2? Answer: 0.7183.

$$SS_y = SS_{reg} + SS_{res} = 808.89 + 317.16 = 1126.05$$

$$r^2 = \frac{SS_{reg}}{SS_y} = \frac{808.89}{1126.05} = 0.7183$$

(b) If $b_1 = -0.041215$ and $s_{b_1} = 0.004712$, what is the value of the test statistic t?

$$t = \frac{b_1}{s_{b_1}} = \frac{-0.041215}{0.004712} = -8.75$$

(c) What is the p-value?

$= 2(p(t < -8.75, df = n - k - 1)) = 2(p(t < -8.75, df = 30)) = 0.00000000093$

```
2 * pt(-8.75, 30)

## [1] 0.0000000009313949
```

(d) Is the estimated regression equation significant at the $\alpha = 0.01$ level?

Yes, since p-value $= 0.00000000093 < \alpha = 0.01$, we conclude that the estimated regression equation is significant.

(e) If $b_0 = 29.599855$, write out the regression equation, $\hat{y} = b_0 + b_1 x$.

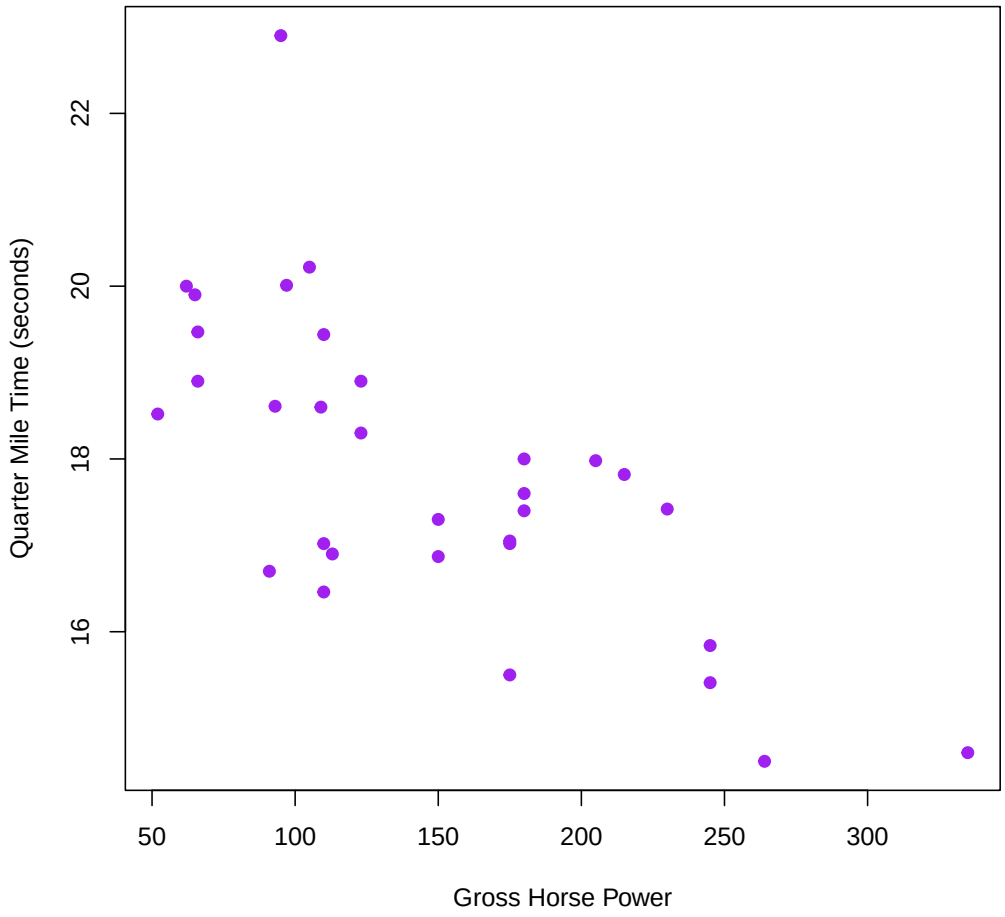
$$\hat{y} = 29.599855 - 0.041215x$$

36. This exercise uses the `mtcars` data set that is part of the basic R installation. (Remember that to see all the installed data sets, simply enter `data()` at the prompt in the Console; to view the `mtcars` data set itself, enter `mtcars` at the prompt; to learn more about the data set, including the variables and observations, enter `?mtcars` at the prompt and wait for the help page to open.) In this case, we are interested in the relationship between an automobile's `quarter mile time` and `gross horsepower`.

(a) Create a scatterplot of the two variables. What does the pattern of points suggest about the relationship (if any) between the variables? Are there any outliers?

Answer: The scatterplot makes clear that the relationship between gross horse power and quarter mile time (seconds) is both negative and (approximately) linear. There appears to be an outlier in the upper lefthand area of the plot.

```
plot(mtcars$hp, mtcars$qsec,
     pch = 19,
     col = 'purple',
     xlab = "Gross Horse Power",
     ylab = "Quarter Mile Time (seconds)")
```



(b) Letting the `quarter mile time` be the dependent variable, estimate the regression equation. Write out the regression equation $\hat{y} = b_0 + b_1 x$.

```
reg_eq_mtcars <- lm(qsec ~ hp, data = mtcars)

reg_eq_mtcars

##
## Call:
## lm(formula = qsec ~ hp, data = mtcars)
##
## Coefficients:
## (Intercept)           hp
##    20.55635     -0.01846
```

The estimated regression equation is: $\hat{y} = 20.55635 - 0.01846x$.

(c) What is the r^2?

```
summary(reg_eq_mtcars)

##
## Call:
## lm(formula = qsec ~ hp, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1766 -0.6975  0.0348  0.6520  4.0972
##
## Coefficients:
##              Estimate Std. Error t value             Pr(>|t|)
## (Intercept) 20.556354   0.542424  37.897 < 0.0000000000000002 ***
## hp          -0.018458   0.003359  -5.495           0.00000577 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.282 on 30 degrees of freedom
## Multiple R-squared:  0.5016,Adjusted R-squared:  0.485
## F-statistic: 30.19 on 1 and 30 DF,  p-value: 0.000005766
```

The r^2 is 0.5016, indicating that approximately 50.16% of variation in the dependent variable is explained by variation in the independent variable.

(d) What is the p-value?

Answer: p-value=0.000005766

(e) Is the estimated regression equation significant at the $\alpha = 0.05$ level?

Since p-value=0.000005766 is less than the usual values for α (such as $\alpha = 0.10$, 0.05, or 0.01), we say that the estimated regression equation is significant.

37. Use the `mtcars` data to answer the following questions.

(a) Find the predicted `quarter mile time` for all the values of `gross horsepower` from the data set used in the original analysis. Report the predicted values for the last four observations.

```
tail(fitted(reg_eq_mtcars) , 4)

## Ford Pantera L   Ferrari Dino  Maserati Bora     Volvo 142E
##       15.68336       17.32615       14.37282       18.54440
```

(b) Find the predicted values of `quarter mile time` for the following values of `gross horsepower`: 100, 125, 160, 225, and 250.

```
new_values <- data.frame(hp <- c(100, 125, 160, 225, 250))

predict(reg_eq_mtcars, new_values)

##        1        2        3        4        5
## 18.71052 18.24906 17.60302 16.40323 15.94178
```

(c) Can we use the estimated regression equation to make predictions of `quarter mile time` when `gross horsepower` is 40 or 350? Why or why not?

Answer: The minimum and maximum values of `hp` are 52 and 335, respectively. Accordingly, we should be very careful about using the estimated regression equation to make predictions based on values that are either above 335 (as is 350) or below 52 (as is 40).

```
min(mtcars$hp)

## [1] 52

max(mtcars$hp)

## [1] 335
```

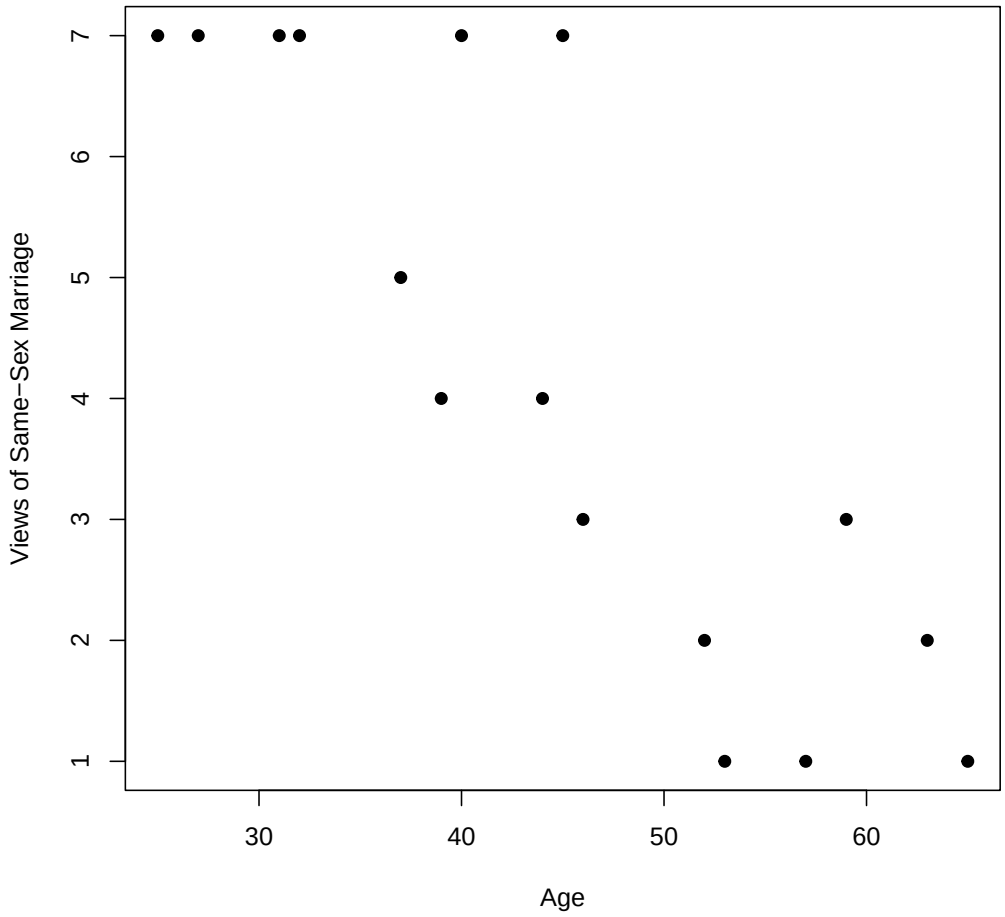
38. This exercise explores the relationship (if any) between two of the five variables of the `polling.csv` data: x_1 =`age`, measured in years, and x_3 =`same sex`, which is measured on a 1-to-7 Likert scale as a response to the statement, "I approve of the right of same-sex couples to marry." A respondent registers strong disapproval with a 1, strong approval with a 7, and relative indifference with a response in the middle of the range from 1-to-7. The `polling.csv` data can be found on the website.

(a) Make a scatterplot of x_3 against x_1. Do you see any possible violations of the assumptions underlying the correct application of simple linear regression to these data? What does the nature of the pattern tell you? Do you think regression can be used to explore the relationship between the two variables?

The scatterplot reveals the negative and (relatively) linear relationship between a person's age and the degree to which she approves of the right of same-sex couples to marry: resistance to the idea that same-sex couples should have the right to marry seems to increase with one's age, although the relationship is not a perfect one. Even so, regression analysis would seem to be a promising means by which to explore the relationship between these two variables.

```
polling <- read.csv('polling.csv') # Import the polling.csv data.
```

```
plot(polling$x1, polling$x3,
     xlab = "Age",
     ylab = "Views of Same-Sex Marriage",
     pch = 19)
```



(b) Write out the regression equation. In this case, does it make more sense to specify x_1 or x_3 as the dependent variable? That is, should you define the model as $x_3 = b_0 + b_1 x_1$? Or as $x_1 = b_0 + b_3 x_3$? Why?

We would most likely specify x_3, `Same-Sex Marriage`, as the dependent variable and x_1, `Age`, as the independent variable. Although we do not use regression analysis to demonstrate causality, it makes more sense to say that approval of the right of same-sex couples to marry falls with age than the reverse.

The estimated regression equation is $\hat{y} = 11.757 - 0.168x$.

```
reg_eq_polling <- lm(x3 ~ x1, data = polling)

reg_eq_polling

##
## Call:
## lm(formula = x3 ~ x1, data = polling)
##
```

```
## Coefficients:
## (Intercept)           x1
##      11.757       -0.168
```

(c) What is r^2?

```
summary(reg_eq_polling)

##
## Call:
## lm(formula = x3 ~ x1, data = polling)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8535 -1.0235 -0.2935  0.6705  2.8025
##
## Coefficients:
##             Estimate Std. Error t value    Pr(>|t|)
## (Intercept)  11.7573     1.2276   9.577 0.000000159 ***
## x1           -0.1680     0.0265  -6.340 0.000018289 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.296 on 14 degrees of freedom
## Multiple R-squared:  0.7417,Adjusted R-squared:  0.7232
## F-statistic: 40.19 on 1 and 14 DF,  p-value: 0.00001829
```

Answer: $r^2 = 0.7417$

(d) What is the p-value?

Answer: p-value=0.00001829

(e) Is the regression equation significant at the $\alpha = 0.05$ level?

Since p-value=0.00001829 is less than the usual values for α (e.g., $\alpha = 0.10$, 0.05, or 0.01), we say that the estimated regression equation is significant.

(f) Find the 95% confidence interval estimate of the regression coefficient β?

```
confint(reg_eq_polling, level = 0.95)

##                  2.5 %     97.5 %
## (Intercept)  9.1242484 14.3903218
## x1          -0.2248278 -0.1111626
```

(g) State in words the meaning of the confidence interval estimate of β.

There is a 95% probability that the regression coefficient falls in the interval from -0.2248278 to -0.1111626.

(h) What are your conclusions about the regression analysis?

Answer: The estimated regression equation $\hat{y} = 11.757 - 0.168x$ allows us to conclude that we can expect that a change of 1 year is associated with a change of -0.168 in approval. (We know this because the regression coefficient is $b_1 = -0.168$.) In this case, the meaning of the intercept term, $b_0 = 11.757$, is less clear because it represents the predicted value of approval for a person whose age is 0. Even so, it is important to retain the intercept term in the equation because it must be included when we want to make predictions.