Exam 1 - Part 2

Neeraj Namani - 001616313

2024-02-25

MPG

This dataset, part of the ggplot2 library, contains a subset of the fuel economy data that the EPA makes available on https://fueleconomy.gov/. It contains only models which had a new release every year between 1999 and 2008 - this was used as a proxy for the popularity of the car.

This data frame has 234 rows and 11 variables (features):

- manufacturer: manufacturer name
- model: model name
- displ: engine displacement, in liters
- year: year of manufacture
- cyl: number of cylinders
- trans: type of transmission
- drv: the type of drive train, where f = front-wheel drive, r = rear wheel drive, 4 = 4wd
- cty: city miles per gallon
- hwy: highway miles per gallon
- fl: fuel type

-class: "type" of car

Let's model mpg in the city.

Look at the type of variables in the dataset

```
library(ggplot2)
str(mpg)
```

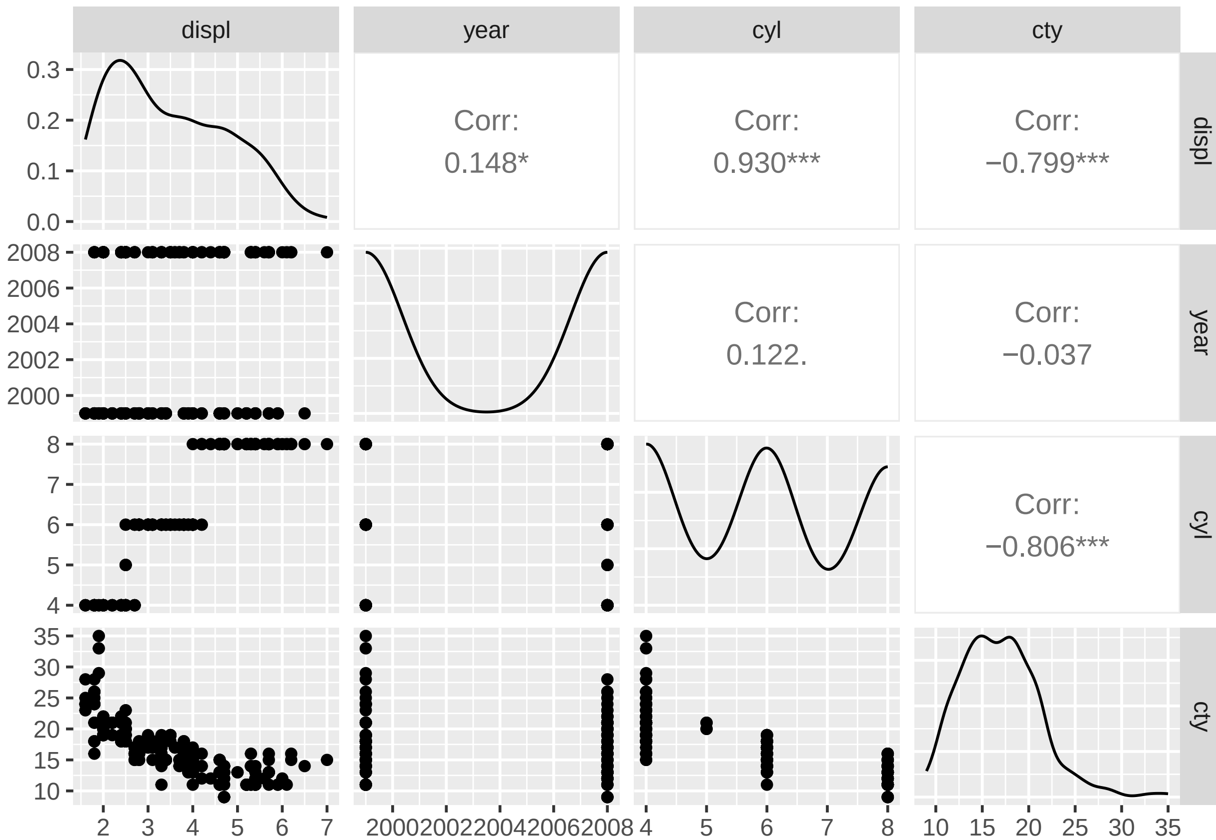
```
## tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
##  $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
##  $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
##  $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr [1:234] "f" "f" "f" "f" ...
##  $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : chr [1:234] "p" "p" "p" "p" ...
##  $ class       : chr [1:234] "compact" "compact" "compact" "compact" ...
```

A. (3) Plot variables that are numeric vs cty. Use ggpairs and select, from mpg, only the numeric variables.

```
library(ggplot2); library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
ggpairs(mpg[,c(3,4,5,8)])
```



Managing variables:

We see that year and cyl are actually categorical. Also, cyl has a few cars with 5 cyl. Maybe we can combine them with the 4 cyl.

```
mympg<-mpg
which(mympg$cyl==5)
```

```
## [1] 218 219 226 227
```

```
mympg$cyl<-ifelse(mympg$cyl==5,4,mympg$cyl)


mympg$cyl<-as.factor(mympg$cyl)
mympg$year<-as.factor(mympg$year)
mympg
```

```
## # A tibble: 234 x 11
##    manufacturer model      displ year  cyl   trans drv     cty   hwy fl    class
##    <chr>        <chr>      <dbl> <fct> <fct> <chr> <chr> <int> <int> <chr> <chr>
##  1 audi         a4           1.8 1999  4     auto~ f        18    29 p     comp~
```

```
##  2 audi         a4           1.8 1999  4     manu~ f         21    29 p     comp~
##  3 audi         a4           2   2008  4     manu~ f         20    31 p     comp~
##  4 audi         a4           2   2008  4     auto~ f         21    30 p     comp~
##  5 audi         a4           2.8 1999  6     auto~ f         16    26 p     comp~
##  6 audi         a4           2.8 1999  6     manu~ f         18    26 p     comp~
##  7 audi         a4           3.1 2008  6     auto~ f         18    27 p     comp~
##  8 audi         a4 quattro   1.8 1999  4     manu~ 4         18    26 p     comp~
##  9 audi         a4 quattro   1.8 1999  4     auto~ 4         16    25 p     comp~
## 10 audi         a4 quattro   2   2008  4     manu~ 4         20    28 p     comp~
## # i 224 more rows
```

B. (3) Consider the following model:cty~displ+year+cyl+trans+drv Run the model and obtain its summary:

```r
model1<- lm(cty ~ displ + year + cyl + trans + drv, data = mympg)
summary(model1)
```

```
##
## Call:
## lm(formula = cty ~ displ + year + cyl + trans + drv, data = mympg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.3589 -1.1215 -0.1582  0.8405 12.8306
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     21.37637    1.42022  15.051  < 2e-16 ***
## displ           -0.91394    0.34448  -2.653 0.008563 **
## year2008         0.46971    0.36673   1.281 0.201629
## cyl6            -3.00587    0.53526  -5.616 5.93e-08 ***
## cyl8            -4.08884    0.99393  -4.114 5.52e-05 ***
## transauto(l3)   -1.08324    1.85211  -0.585 0.559242
## transauto(l4)   -1.09142    1.04415  -1.045 0.297057
## transauto(l5)   -1.05275    1.05704  -0.996 0.320382
## transauto(l6)   -1.75935    1.32482  -1.328 0.185569
## transauto(s4)    0.44600    1.59417   0.280 0.779917
## transauto(s5)   -0.45611    1.57698  -0.289 0.772682
## transauto(s6)   -0.57330    1.11571  -0.514 0.607881
## transmanual(m5) -0.09659    1.05552  -0.092 0.927170
## transmanual(m6) -0.79386    1.09876  -0.723 0.470760
## drvf             2.62614    0.39099   6.717 1.59e-10 ***
## drvr             2.00094    0.53021   3.774 0.000207 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.15 on 218 degrees of freedom
## Multiple R-squared:  0.7613, Adjusted R-squared:  0.7448
## F-statistic: 46.34 on 15 and 218 DF,  p-value: < 2.2e-16
```

C. (2) How many categories does the variable "trans" have?

```r
unique_trans <- (mpg$trans)
table(unique_trans)
```

```
## unique_trans
##   auto(av)   auto(l3)   auto(l4)   auto(l5)   auto(l6)   auto(s4)   auto(s5)
```

```
##          5          2         83         39          6          3          3
##   auto(s6) manual(m5) manual(m6)
##         16         58         19
```

Ans. We have 10 categories in the "trans".

D. (2) How many categories does the variable "drv" have? Find what they are.

```r
unique_drv <- (mpg$drv)
table(unique_drv)
```

```
## unique_drv
##   4   f   r
## 103 106  25
```

Ans. We have 3 categories in the "drv". They are 4 - Four wheel drive, f - front wheel drive, r - rear wheel drive.

E. (2) We do not loose much by removing trans. Consider model2: cty~displ+year+cyl+drv. Run the model and find its summary.

```r
model2<-lm(cty ~ displ + year + cyl + drv, data = mympg)
summary(model2)
```

```
##
## Call:
## lm(formula = cty ~ displ + year + cyl + drv, data = mympg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.6659 -1.0895 -0.0230  0.9704 13.2803
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  21.0362     0.8855  23.757  < 2e-16 ***
## displ        -0.9945     0.3375  -2.947 0.003542 **
## year2008      0.4797     0.2872   1.670 0.096218 .
## cyl6         -3.1411     0.5217  -6.021 6.90e-09 ***
## cyl8         -4.3035     0.9795  -4.393 1.71e-05 ***
## drvf          2.5731     0.3750   6.861 6.43e-11 ***
## drvr          2.0404     0.5197   3.926 0.000114 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.154 on 227 degrees of freedom
## Multiple R-squared:  0.7504, Adjusted R-squared:  0.7439
## F-statistic: 113.8 on 6 and 227 DF,  p-value: < 2.2e-16
```

F. (4) Interpret the effect of cyl on mpg.

Ans. From the summary,

"cyl6": The coefficient for 6-cylinder cars is -3.1411. This means that, 6-cylinder cars are expected to have a city mpg that is 3.1411 units lower than the baseline category of cylinders (which is likely 4-cylinder cars, since it's not listed). The p-value is extremely small (6.90e-09), indicating that this effect is statistically significant.

"cyl8": The coefficient for 8-cylinder cars is -4.3035. This implies that 8-cylinder cars are expected to have a city mpg that is 4.3035 units lower than the baseline category of cylinders, again likely 4-cylinder cars, holding all other variables constant. The p-value here is also very small (1.71e-05), suggesting that the effect of having 8 cylinders compared to the baseline is statistically significant.

G. (3) Interpret the effect of displ on mpg.

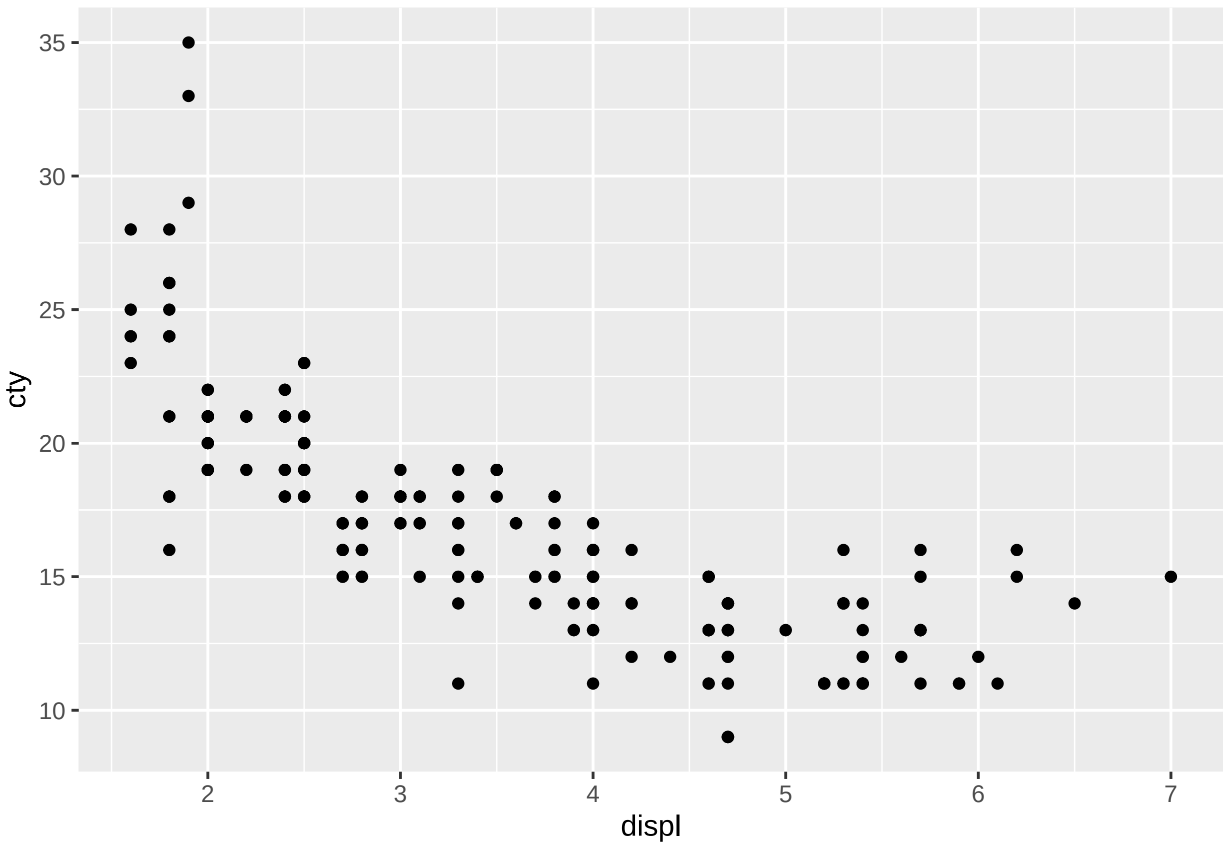
Ans. The coefficient for engine displacement (displ) in the model2 summary is -0.9945. This number represents the change in city miles per gallon (cty) for each one-unit increase in engine displacement, measured in liters, holding all other variables constant.

Negative Relationship: The negative sign of the coefficient (-0.9945) indicates a negative relationship between engine displacement and city mpg. This means that as the engine displacement increases, the city mpg tends to decrease.

Magnitude of Effect: Specifically, for each one-liter increase in engine displacement, the city mpg is expected to decrease by approximately 0.9945 miles per gallon. This effect is statistically significant, as suggested by the p-value (0.003542), which is less than the conventional alpha level of 0.05. The significance is also denoted by the two stars next to the coefficient, indicating a confidence level of 99% (or a significance level of 0.01).

H. (2) Now consider the following graph:

```r
ggplot(mympg,aes(x=displ,y=cty))+geom_point()
```




```r
  geom_smooth(method = lm)
```

```
## geom_smooth: na.rm = FALSE, orientation = NA, se = TRUE
## stat_smooth: na.rm = FALSE, orientation = NA, se = TRUE, method = function (formula, data, subset, we
## {
##     ret.x <- x
##     ret.y <- y
##     cl <- match.call()
```

```
##     mf <- match.call(expand.dots = FALSE)
##     m <- match(c("formula", "data", "subset", "weights", "na.action", "offset"), names(mf), 0)
##     mf <- mf[c(1, m)]
##     mf$drop.unused.levels <- TRUE
##     mf[[1]] <- quote(stats::model.frame)
##     mf <- eval(mf, parent.frame())
##     if (method == "model.frame")
##         return(mf)
##     else if (method != "qr")
##         warning(gettextf("method = '%s' is not supported. Using 'qr'", method), domain = NA)
##     mt <- attr(mf, "terms")
##     y <- model.response(mf, "numeric")
##     w <- as.vector(model.weights(mf))
##     if (!is.null(w) && !is.numeric(w))
##         stop("'weights' must be a numeric vector")
##     offset <- model.offset(mf)
##     mlm <- is.matrix(y)
##     ny <- if (mlm)
##         nrow(y)
##     else length(y)
##     if (!is.null(offset)) {
##         if (!mlm)
##             offset <- as.vector(offset)
##         if (NROW(offset) != ny)
##             stop(gettextf("number of offsets is %d, should equal %d (number of observations)", NROW(o
##     }
##     if (is.empty.model(mt)) {
##         x <- NULL
##         z <- list(coefficients = if (mlm) matrix(NA, 0, ncol(y)) else numeric(), residuals = y, fitte
##         if (!is.null(offset)) {
##             z$fitted.values <- offset
##             z$residuals <- y - offset
##         }
##     }
##     else {
##         x <- model.matrix(mt, mf, contrasts)
##         z <- if (is.null(w))
##             lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...)
##         else lm.wfit(x, y, w, offset = offset, singular.ok = singular.ok, ...)
##     }
##     class(z) <- c(if (mlm) "mlm", "lm")
##     z$na.action <- attr(mf, "na.action")
##     z$offset <- offset
##     z$contrasts <- attr(x, "contrasts")
##     z$xlevels <- .getXlevels(mt, mf)
##     z$call <- cl
##     z$terms <- mt
##     if (model)
##         z$model <- mf
##     if (ret.x)
##         z$x <- x
##     if (ret.y)
##         z$y <- y
##     if (!qr)
```

```
##         z$qr <- NULL
##     z
## }
## position_identity
```

Update model2 by replacing displ with log(displ)

```r
model3<-lm(cty ~ log(displ) + year + cyl + drv, data = mympg)
summary(model3)
```

```
##
## Call:
## lm(formula = cty ~ log(displ) + year + cyl + drv, data = mympg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5883 -1.0651 -0.0143  1.1305 13.0617
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  23.2106     1.0257  22.628  < 2e-16 ***
## log(displ)   -5.4182     1.1449  -4.732 3.90e-06 ***
## year2008      0.6137     0.2816   2.180  0.03031 *
## cyl6         -1.9726     0.5941  -3.320  0.00105 **
## cyl8         -2.8406     0.9569  -2.969  0.00331 **
## drvf          2.2054     0.3773   5.845 1.75e-08 ***
## drvr          2.0704     0.4969   4.167 4.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.094 on 227 degrees of freedom
## Multiple R-squared:  0.7642, Adjusted R-squared:  0.7579
## F-statistic: 122.6 on 6 and 227 DF,  p-value: < 2.2e-16
```

I. (4) Make a scatterplot with x=displ and y=cty. Overlay a curve with the fitted values from model3.

```r
library(ggplot2)

p <- ggplot(mympg, aes(x = displ, y = cty)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ log(x), se = FALSE,color = "red") +
  theme_minimal() +
  labs(x = "Engine Displacement", y = "City",
       title  = "(City vs. Engine Displacement) with Linear-Log Regression Line")

print(p)
```

J. (4) Interpret the coefficient of log(displ) in model3.

Look at the notes on how to interpret when you use logarithms.

Ans. The coefficient for log(displ) in model3 is -5.4182. This model uses the natural logarithm of engine displacement (displ) as a predictor for city miles per gallon (cty). The coefficient represents the change in cty associated with a 1-unit change in the natural log of displ, holding all other variables constant.

Negative Relationship: The negative coefficient indicates that as engine displacement increases, city mpg decreases, but the relationship is logarithmic rather than linear. This means the effect of increasing displ on cty diminishes as displ increases. A larger engine displacement has a greater negative impact on fuel efficiency, but each additional liter of displacement has a progressively smaller effect on reducing city mpg when considered on a logarithmic scale.

Magnitude of Effect: For a 1-unit increase in the natural logarithm of displ, the city mpg is expected to decrease by 5.4182 units. Given the properties of the natural logarithm, this doesn't translate to a straightforward "per liter" decrease in mpg as with a linear term. Instead, the impact of an increase in displ is dependent on the current value of displ; percentage-wise increases in displ have consistent effects on cty.

Statistical Significance: The p-value associated with the log(displ) coefficient is extremely small (3.90e-06), indicating that the relationship between the logarithm of engine displacement and city mpg is statistically significant. This suggests that the logarithmic transformation of displ provides a meaningful and significant explanation of variations in cty.

K. (2) Update model2 by replacing displ with a polynomial on "dspl" of degree 2:

```
model4<-lm(cty ~ poly(displ, 2) + year + cyl + drv, data = mympg)
summary(model4)
```

```
##
## Call:
```

```
## lm(formula = cty ~ poly(displ, 2) + year + cyl + drv, data = mympg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5140 -1.0115  0.0036  0.9541 12.9250
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      16.6533     0.5361  31.066  < 2e-16 ***
## poly(displ, 2)1 -30.8049     6.7712  -4.549 8.78e-06 ***
## poly(displ, 2)2  14.4693     3.0215   4.789 3.03e-06 ***
## year2008          0.7044     0.2782   2.532  0.01203 *
## cyl6             -1.1495     0.6489  -1.771  0.07784 .
## cyl8             -2.7355     0.9910  -2.760  0.00625 **
## drvf              1.9519     0.3809   5.125 6.39e-07 ***
## drvr              1.6447     0.5031   3.269  0.00125 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.057 on 226 degrees of freedom
## Multiple R-squared:  0.7734, Adjusted R-squared:  0.7664
## F-statistic: 110.2 on 7 and 226 DF,  p-value: < 2.2e-16
```

L. (4) Select the model that fits better according to the size of R^2. Look at the assessment plots.

Model: Model4 has the highest $R^2 = 0.7734374$

Plots:

```r
rSquared_model1 <- summary(model1)$r.squared
rSquared_model2 <- summary(model2)$r.squared
rSquared_model3 <- summary(model3)$r.squared
rSquared_model4 <- summary(model4)$r.squared
cat("The R-Square value for the model1 is: ",rSquared_model1,"\n")
```

```
## The R-Square value for the model1 is:  0.761259
```

```r
cat("The R-Square value for the model2 is: ",rSquared_model2,"\n")
```

```
## The R-Square value for the model2 is:  0.7504477
```

```r
cat("The R-Square value for the model3 is: ",rSquared_model3,"\n")
```

```
## The R-Square value for the model3 is:  0.7641663
```

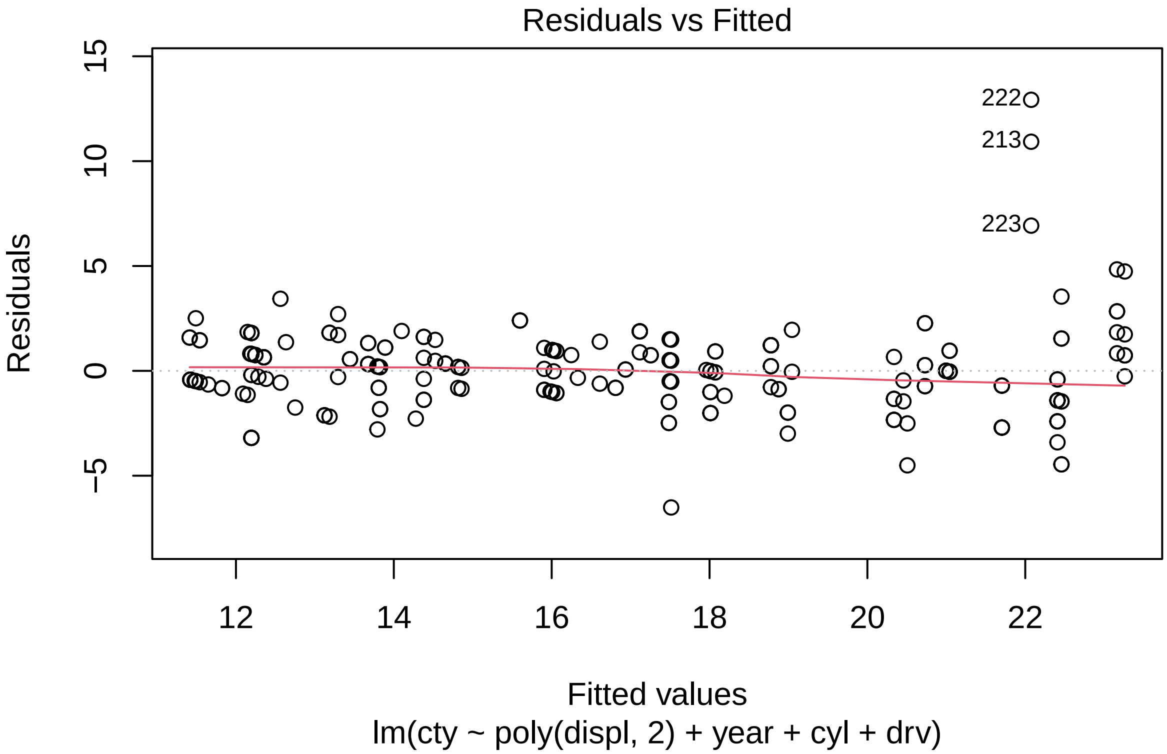
```r
cat("The R-Square value for the model4 is: ",rSquared_model4,"\n")
```

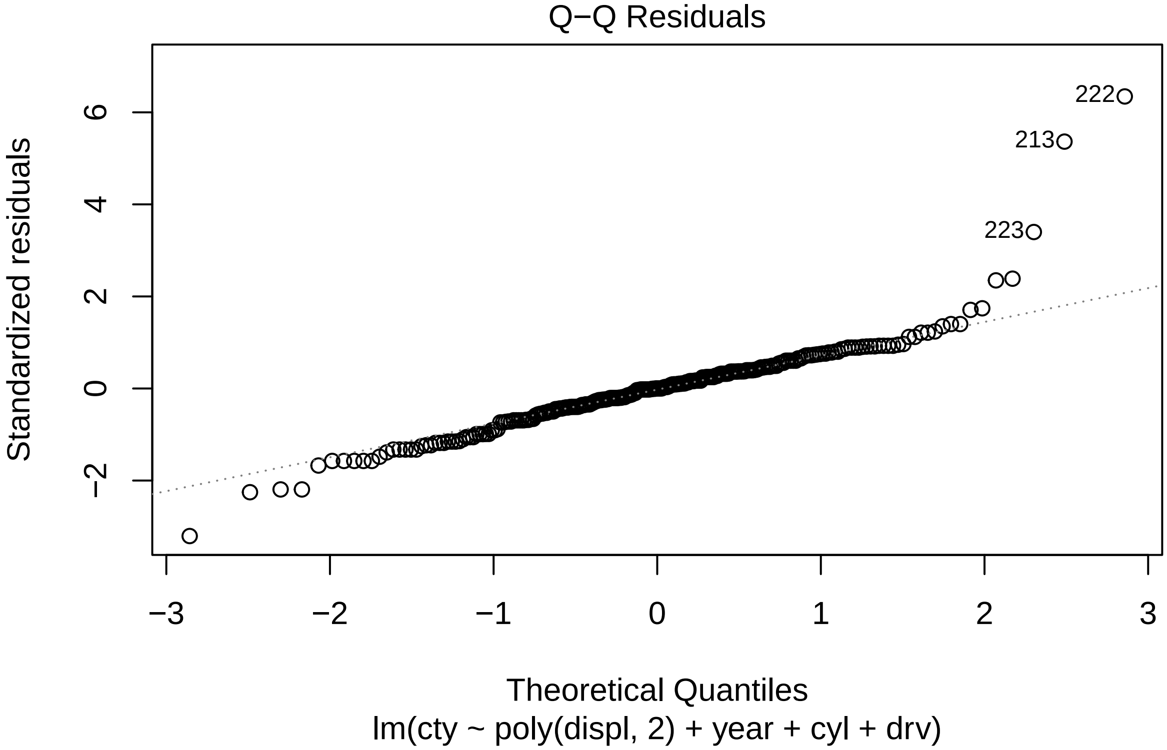
```
## The R-Square value for the model4 is:  0.7734374
```

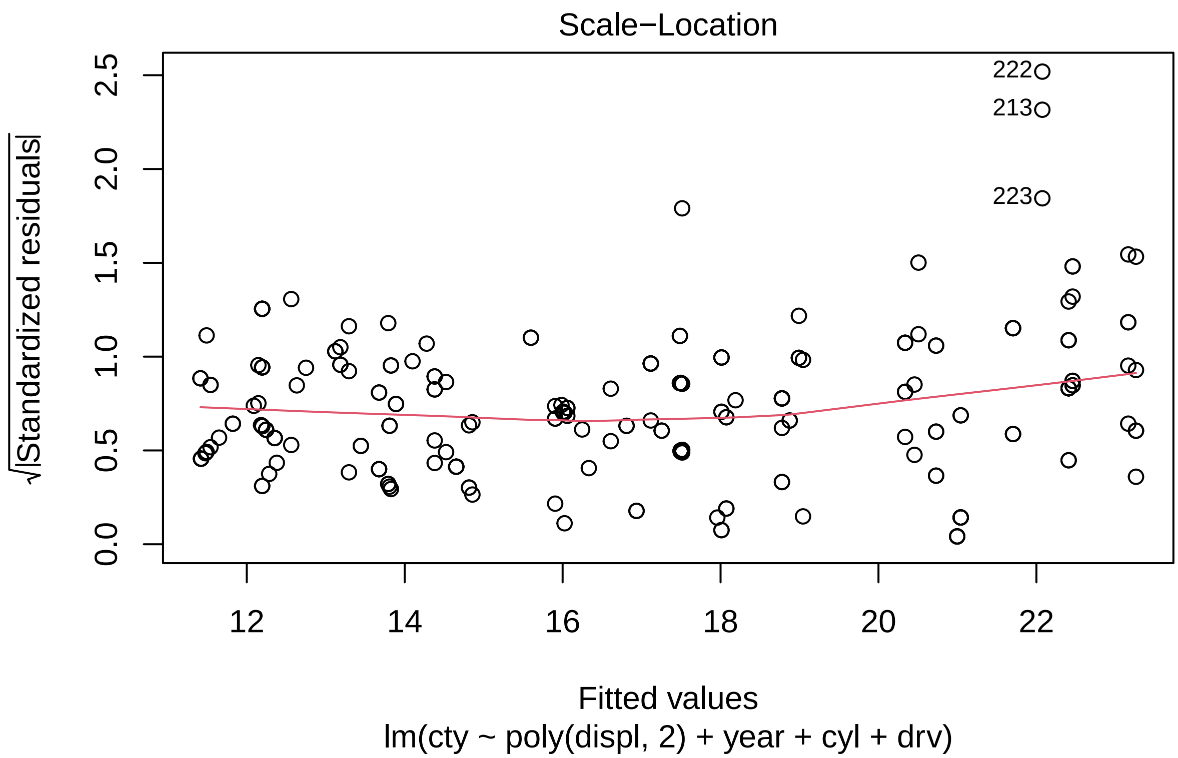
```r
highestrSquaredModel <- max(rSquared_model1, rSquared_model2, rSquared_model3, rSquared_model4)
cat("The highest R-Square value for the model is: ", highestrSquaredModel)
```

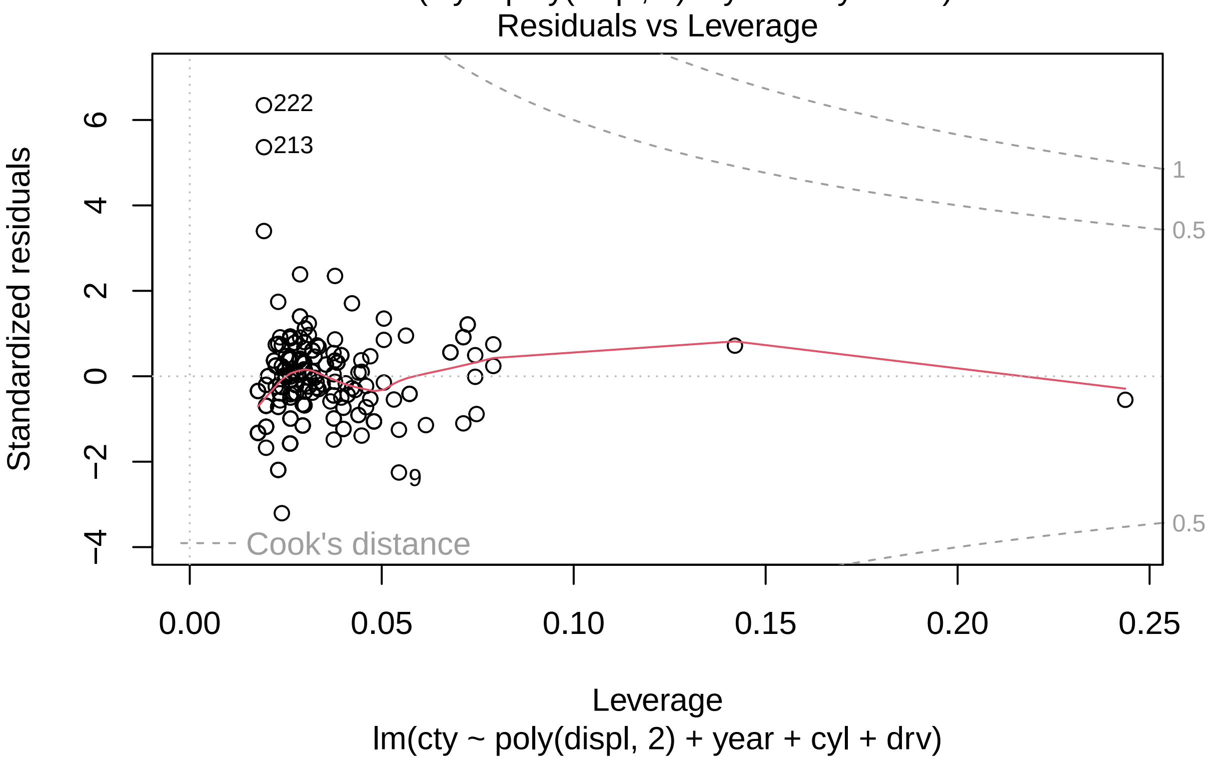
```
## The highest R-Square value for the model is:  0.7734374
```

```r
#par(mfrow = c(2, 2))
plot(model4)
```


Residuals vs Fitted
Residuals
Fitted values
lm(cty ~ poly(displ, 2) + year + cyl + drv)
222
213
223


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(cty ~ poly(displ, 2) + year + cyl + drv)
222
213
223





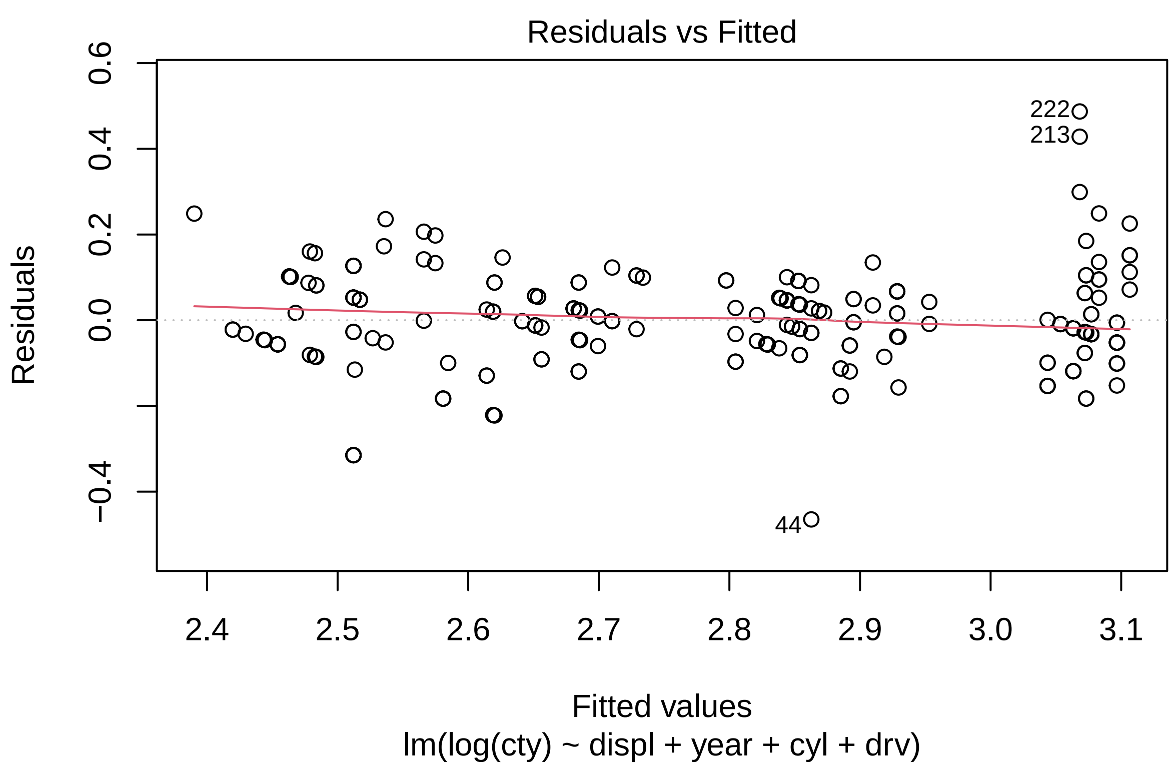
M. (3) Now try one more model by taking the model in E. and using log(cty) instead of cty. Print out the summary of the model and look at the assessment plots.

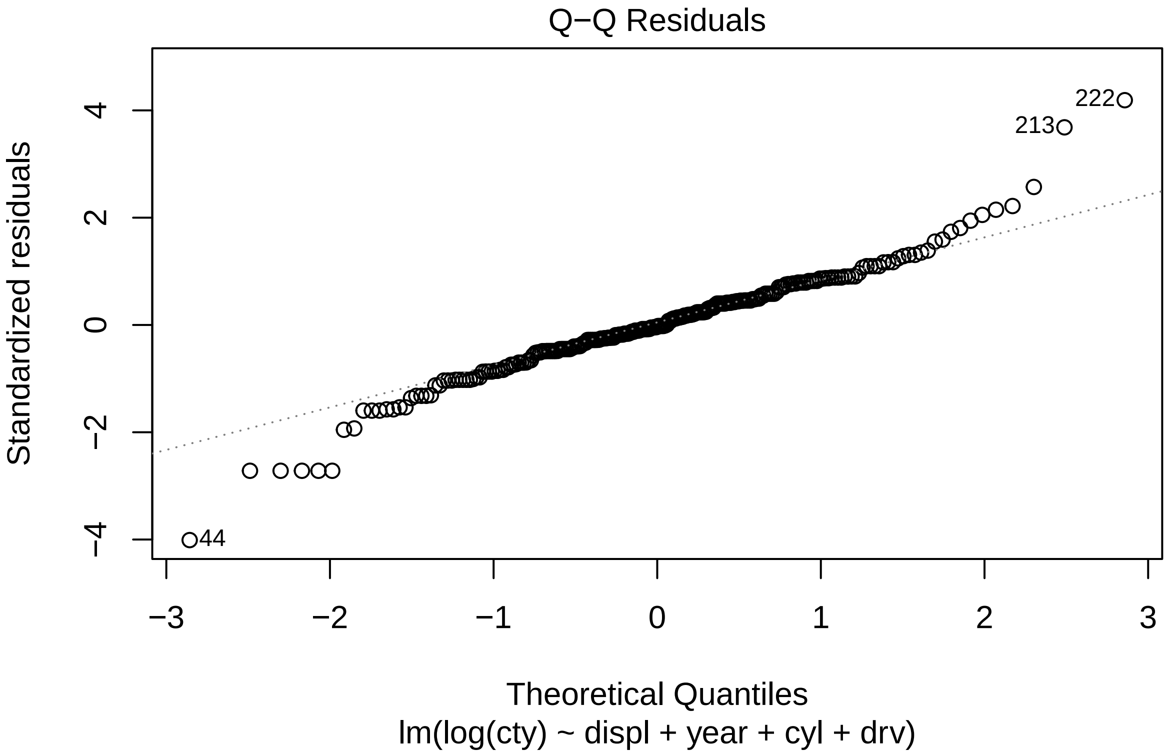
```
model5 <- lm(log(cty) ~ displ + year + cyl + drv, data = mympg)
summary(model5)
```

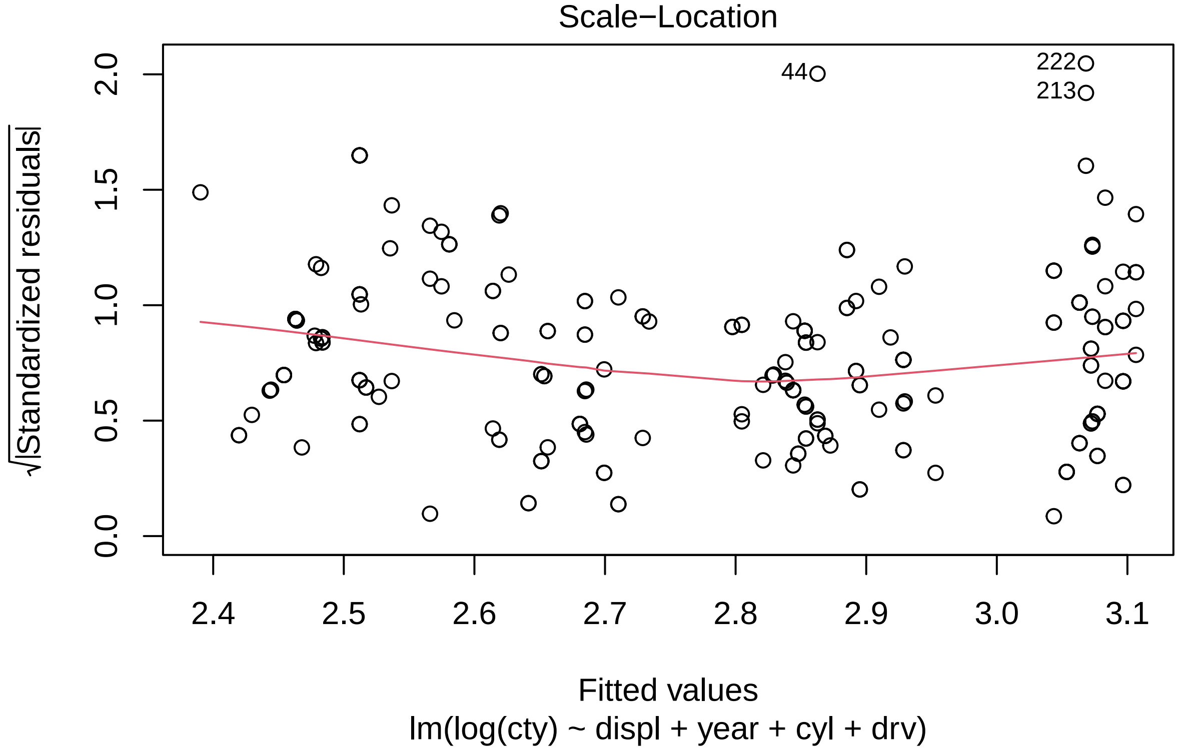
```
##
```

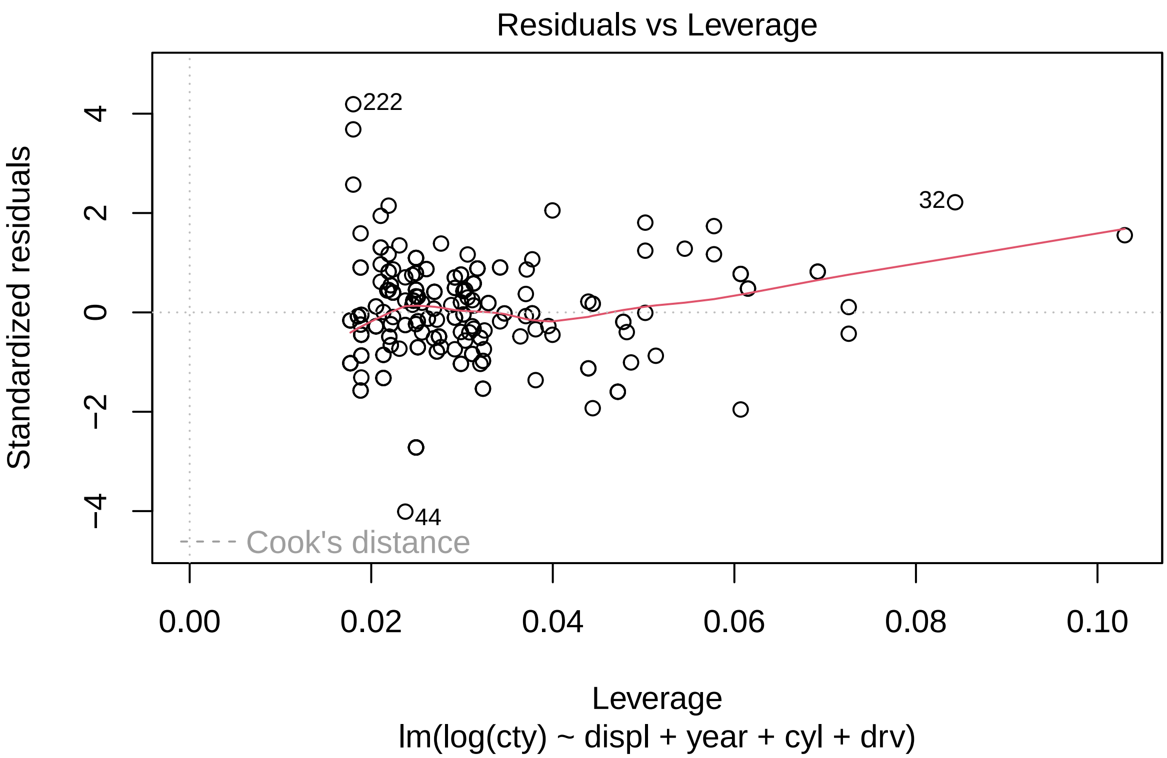
```
## Call:
## lm(formula = log(cty) ~ displ + year + cyl + drv, data = mympg)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46481 -0.05626 -0.00353  0.06726  0.48710
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.01802    0.04822  62.585  < 2e-16 ***
## displ       -0.04918    0.01838  -2.676   0.0080 **
## year2008     0.03339    0.01564   2.135   0.0338 *
## cyl6        -0.17009    0.02841  -5.987 8.28e-09 ***
## cyl8        -0.30816    0.05335  -5.777 2.49e-08 ***
## drvf         0.14366    0.02042   7.034 2.34e-11 ***
## drvr         0.13645    0.02830   4.821 2.61e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1173 on 227 degrees of freedom
## Multiple R-squared:  0.7841, Adjusted R-squared:  0.7784
## F-statistic: 137.4 on 6 and 227 DF,  p-value: < 2.2e-16
```

```r
#par(mfrow=c(2,2))
plot(model5)
```




Q−Q Residuals
Standardized residuals
222
213
44
Theoretical Quantiles
lm(log(cty) ~ displ + year + cyl + drv)


Scale−Location
√|Standardized residuals|
44
222
213
Fitted values
lm(log(cty) ~ displ + year + cyl + drv)



N. (2) Write a sentence that summarizes your thoughts on the models chosen in E and F.

Ans. Model E, which predicts city miles per gallon (`cty`) using engine displacement (`displ`), manufacture year (`year`), number of cylinders (`cyl`), and drive (`drv`), demonstrates a strong relationship between these predictors and city fuel efficiency. The significant coefficients for `cyl6` and `cyl8` indicate a clear decrease in `cty` as the number of cylinders increases, suggesting that engine size and configuration are important determinants of fuel consumption in city driving. The model's high R^2 value of 0.7504 indicates that a substantial portion of the variability in city mpg is explained by these factors, highlighting the model's effectiveness in capturing the underlying patterns in the data.

In E and F, we have chosen the linear model lm(formula = cty ~ displ + year + cyl + drv, data = mympg). But in F, we have interpreted the effect of cyl on mpg.