Analysis of the Mtcars Dataset

Jon Kinsey

Wed Dec 3 13:29:56 2014

```
# This is an analysis of the mtcars dataset which was extracted from the
# 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects
# of automobile design and performance for 32 automobiles (1973-74 models).
# A markup of the report is in mtcars1.pdf created using "Compile Notebook"
#
# Summary
#
# * The mpg is largely determined by the interplay between weight,
#   acceleration, and transmission type.
# * On average, AT cars consume for gasoline than MT cars.
#   The avg mpg for AT cars = 24.4 and for MT cars = 17.2 mpg.
# * the adjusted estimate for the expected change in mpg
#   going from AT to MT is +2.94 gallons.
# * This estimation has a confidence interval of [3.2, 11.3]

data(mtcars)
str(mtcars)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```
#
# where we have:
# [, 1]   mpg    Miles/(US) gallon
# [, 2]  cyl     Number of cylinders
# [, 3]  disp    Displacement (cu.in.)
# [, 4]  hp  Gross horsepower
# [, 5]  drat    Rear axle ratio
# [, 6]  wt  Weight (lb/1000)
# [, 7]  qsec    1/4 mile time
# [, 8]  vs  V/S
# [, 9]  am  Transmission (0 = automatic, 1 = manual)
# [,10]  gear    Number of forward gears
# [,11]  carb    Number of carburetors
#
# look at gas guzzlers using OR
subset(mtcars, mpg < 14 | disp > 390)
```

```
##                      mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Cadillac Fleetwood  10.4   8  472 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8  460 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8  440 230 3.23 5.345 17.42  0  0    3    4
## Camaro Z28          13.3   8  350 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8  400 175 3.08 3.845 17.05  0  0    3    2
```

```
#
# Correlation Analysis
```

```
# Before we perform a regression analysis lets examine the correlation
# between mpg and the other 10 variables using the cor() function.
#
names(mtcars)
```

```
##  [1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
## [11] "carb"
```

```
x <- mtcars[1]
y <- mtcars[2:10]
cor(x,y)
```

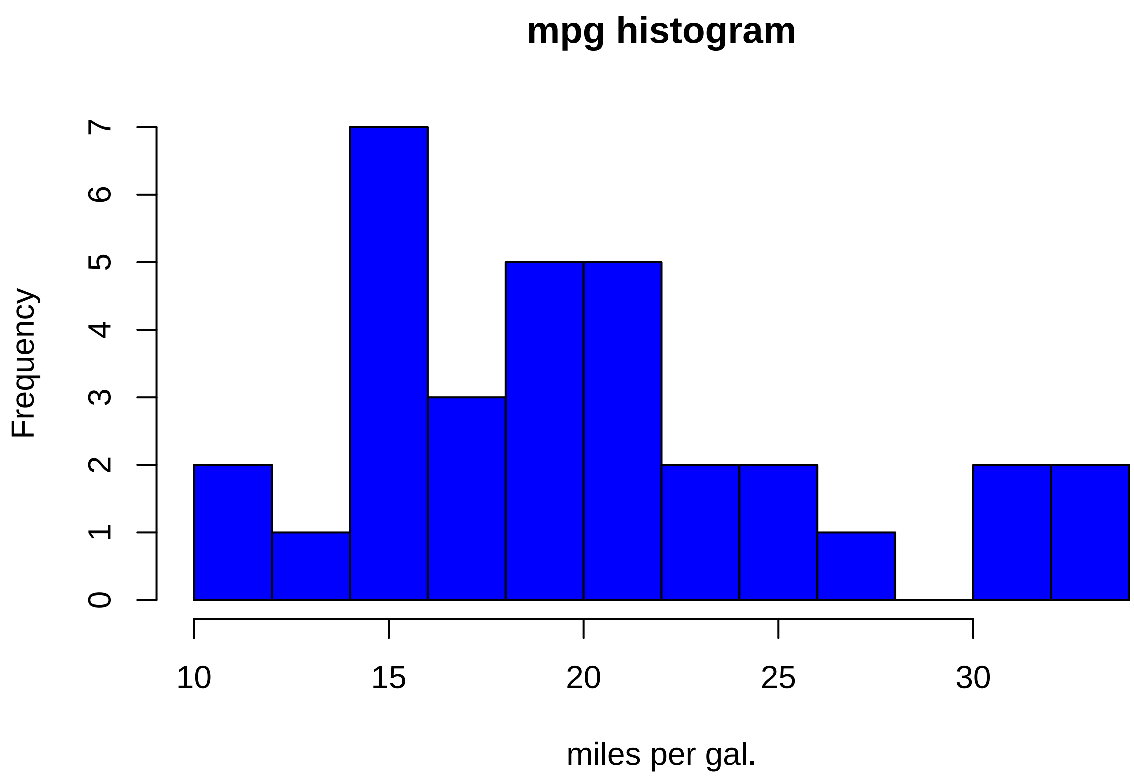
```
##         cyl    disp      hp   drat      wt   qsec    vs     am   gear
## mpg -0.8522 -0.8476 -0.7762 0.6812 -0.8677 0.4187 0.664 0.5998 0.4803
```

```
# Here, we see that cyl, hp, wt, and carb are all negatively correlated with mpg.
# Details of the correlation test between mpg and wt using the Pearson's
# product-moment correlation:
cor.test(mtcars$mpg, mtcars$wt)
```

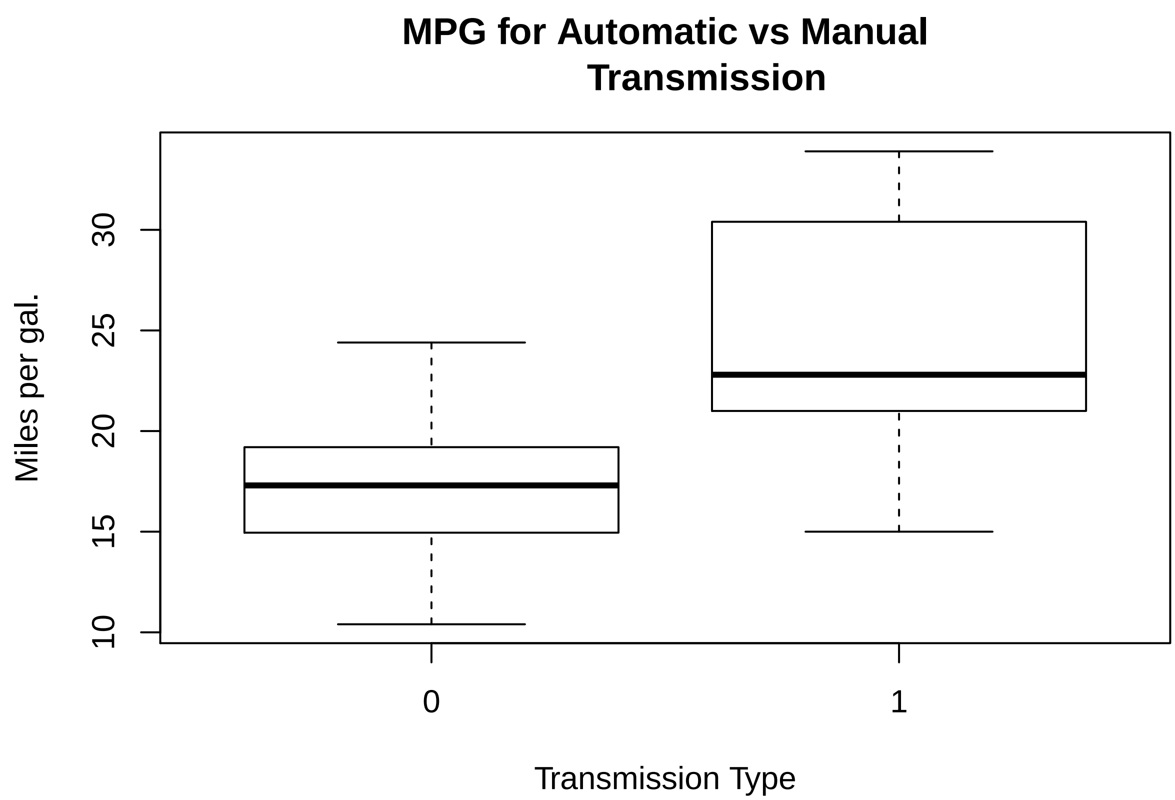
```
##
##  Pearson's product-moment correlation
##
## data:  mtcars$mpg and mtcars$wt
## t = -9.559, df = 30, p-value = 1.294e-10
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.9338 -0.7441
## sample estimates:
##     cor
## -0.8677
```

```
# Histogram plots

# Fig.1 frequency vs mpg
# This shows that the distribution resembles a normal distribution
par(mfrow=c(1,1))
x <- mtcars$mpg
hist(x, breaks=10, xlab="miles per gal.",
     main="mpg histogram", col="blue")
```




```
#
# Histogram with a normal curve ... couldn't get to work
# data(mtcars)
# par(mfrow=c(1,2))
# x <- mtcars$mpg
# h <-hist(x, breaks=10, xlab="miles per gal.",
#      main="mpg histogram", col="red")
# xfit <- seq(min(x), max(x), length=40)
# yfit <- dnorm(xfit,mean=mean(x), sd=sd(x))
# yfit <- yfit*diff*(h$mids[1:2])*length(x)
# lines(xfit,yfit,col="blue", lwd=2)
#
# Fig.2 MPG for Auto vs Manual Transmission
boxplot(mpg ~ am, data=mtcars, xlab="Transmission Type",
        ylab="Miles per gal.", main="MPG for Automatic vs Manual
        Transmission")
```




```
#
# Hypothesis Testing and t-test
#
# First, we convert am from numberical into categorical
mtcars$am <- as.factor(mtcars$am)
levels(mtcars$am) <- c("Automatic", "Manual")
aggregate(mpg ~ am, data=mtcars, mean)
```

```
##          am   mpg
## 1 Automatic 17.15
## 2    Manual 24.39
```

```
#
# To get exact values and confidence intervals for fuel consumption
# of AT vs MT vehicles, we need to split the dataset for AT and MT
# separately and then apply the t-test
#
mtcars$am <- as.factor(mtcars$am)
levels(mtcars$am) <- c("AT", "MT")
mpg.at <- mtcars[mtcars$am == "AT",]$mpg
mpg.mt <- mtcars[mtcars$am == "MT",]$mpg
t.test(mpg.at, mpg.mt)
```

```
##
##  Welch Two Sample t-test
##
## data:  mpg.at and mpg.mt
```

```
## t = -3.767, df = 18.33, p-value = 0.001374
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -11.28  -3.21
## sample estimates:
## mean of x mean of y
##     17.15     24.39
```

```r
#
# Here, we see that the p-value=0.001374 is much less than 0.05
# so we can reject the null hypothesis and conclude AT cars
# have a lower mpg than MT cars. This is based on the assumption
# that all other characteristics of AT and MT cars are the same (e.g same
# weight distribution). In any case, the alternative hypothesis is true, so
# the difference in the means is not equal to zero. Indeed we see that the
# mean for AT's is 17.2 mpg and for MT's it is 24.4 mpg. The 95% confidence
# interval of the difference in the mean mpg is between 3.21 and 11.28 mpg.
# From this we can conclude that MT's are better than AT's in terms of MPG
#
#
# Regression Analysis
```

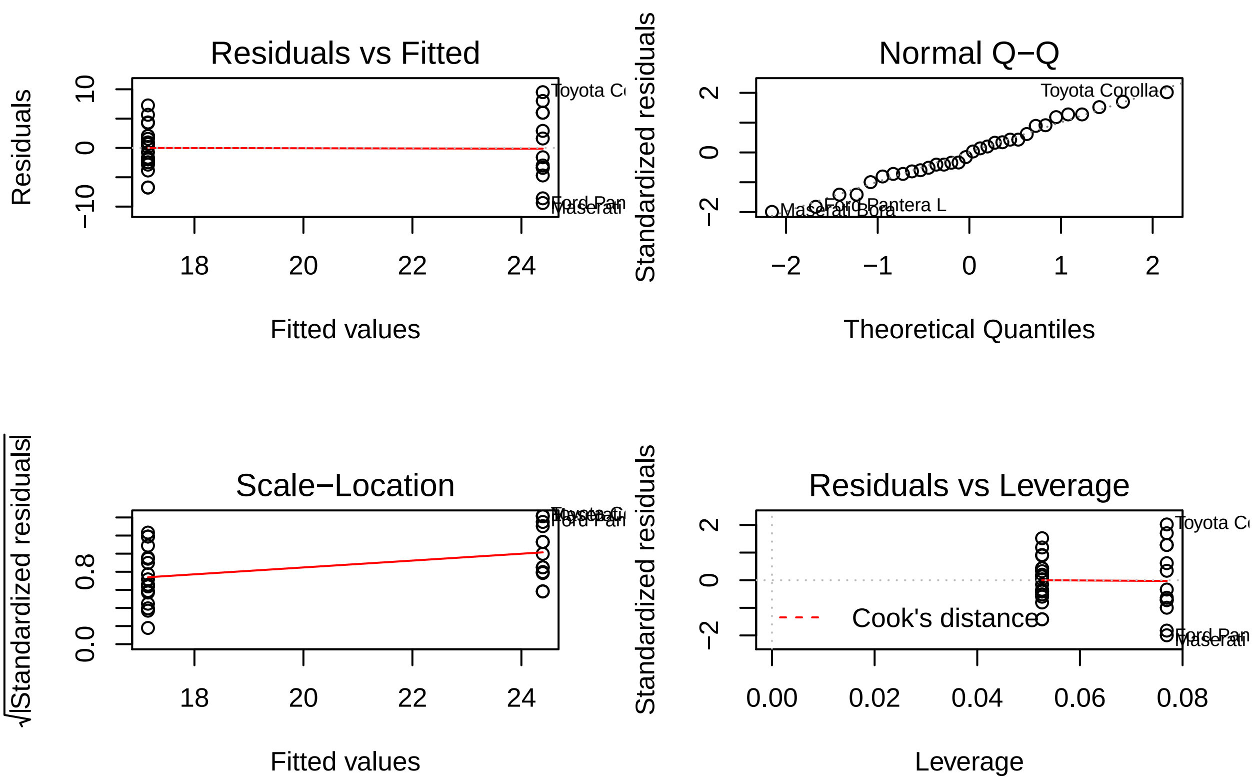
```r
# Single variable Analysis
# First, we begin with only one predictor, am
# Recall that the tilde means "explained by"
# The "Estimate" is an estimation of the slope, below that is the coeff.
# or slope of the weight. If thats positive then increasing that variable
# increases the mpg. Negative means a decrease in mpg. The std error represents
# the amount of uncertainty in our estimate of the slope. The 3rd column has
# the test statistic or t-value. The last column has the p-value which
# describes whether the relationship could be due to chance alone.
#
model1 <- lm(mpg ~ am, data=mtcars)
summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ am, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.392 -3.092 -0.297  3.244  9.508
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    17.15       1.12   15.25  1.1e-15 ***
## amMT            7.24       1.76    4.11  0.00029 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.9 on 30 degrees of freedom
## Multiple R-squared:  0.36,   Adjusted R-squared:  0.338
## F-statistic: 16.9 on 1 and 30 DF,  p-value: 0.000285
```

```r
#
#
# A cleaner look at the coefficients:
summary(model1)$coefficients
```

```
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)   17.147      1.125  15.247 1.134e-15
## amMT           7.245      1.764   4.106 2.850e-04
```

```r
#
# Since the p-value=0.000285 is much less than 0.05 we can
# reject the null hypothesis but we note that the regression
# model only covers 36% (mult. R-squared) of the variance.
# The model coefficients are:
#   intercept = 17.15 represents AT cars mean mpg
#   am coefficient = 7.24 represents the adjusted estimate
#                    for the expected change in mpg comparing
#                    AT vs MT. In other words, a MT should be
#                    expected to have a mpg increase of 7.2 mpg.
# The t-value and residual variation implies a poor fit of the single
# variable model. This is evident in the residual plots (see Fig.3)
par(mfrow = c(2,2))
plot(model1)
```




```r
#
# Multi-variable Analysis
#
```

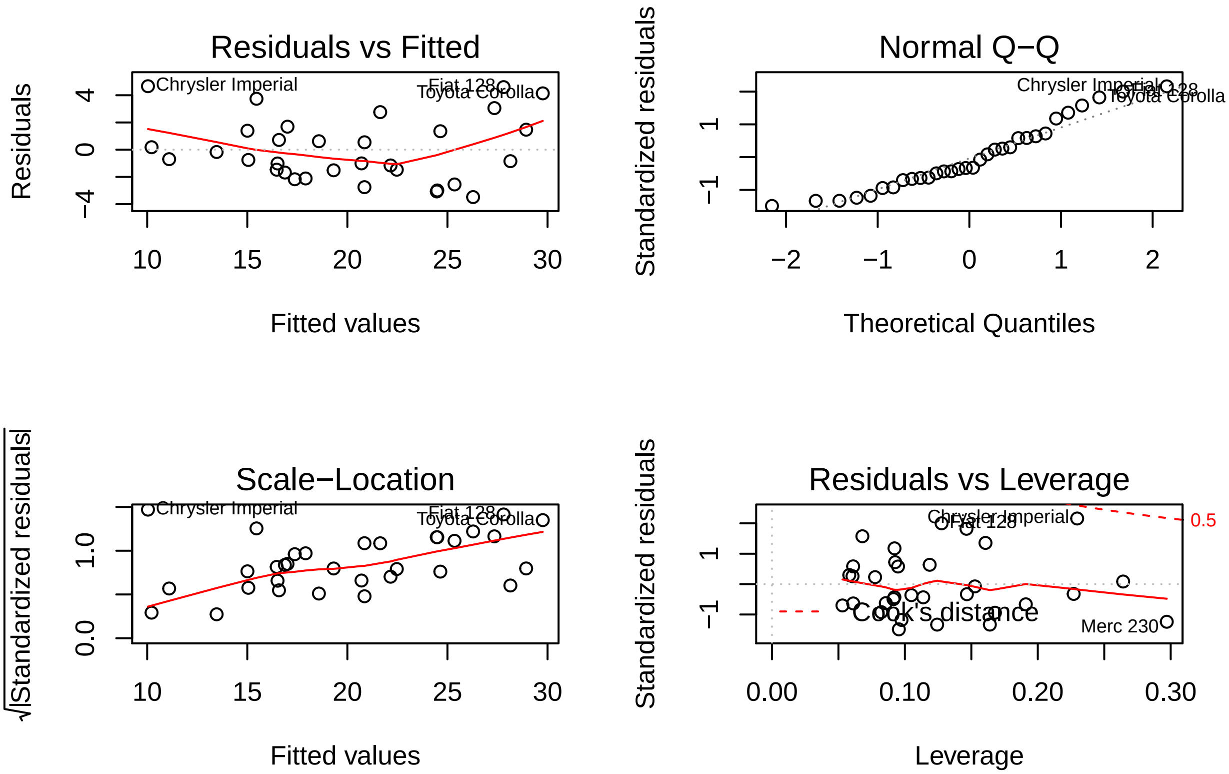
```r
# Next, we use a more complex regression model which contains
# all the independent variables as predictors.
model.all <- lm(mpg ~ ., data=mtcars)
# Now, go backwards to the model that fits the best using the
# a stepwise algorithm
model.best <- step(model.all, trace=0)
summary(model.best)
```

```
##
## Call:
## lm(formula = mpg ~ wt + qsec + am, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.481 -1.556 -0.726  1.411  4.661
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    9.618      6.960    1.38  0.17792
## wt            -3.917      0.711   -5.51    7e-06 ***
## qsec           1.226      0.289    4.25  0.00022 ***
## amMT           2.936      1.411    2.08  0.04672 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.46 on 28 degrees of freedom
## Multiple R-squared:  0.85,   Adjusted R-squared:  0.834
## F-statistic: 52.7 on 3 and 28 DF,  p-value: 1.21e-11
```

```r
#
#
summary(model.best)$coefficients
```

```
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)    9.618     6.9596   1.382 1.779e-01
## wt            -3.917     0.7112  -5.507 6.953e-06
## qsec           1.226     0.2887   4.247 2.162e-04
## amMT           2.936     1.4109   2.081 4.672e-02
```

```r
#
# This model explains 85% of the mpg variance and contains
# only 3 predictors with a formula of mpg ~ wt + qsec + am
#
# The estimated coeff. for amMT is 2.936 and represents the
# adjusted estimate for the expected change in mpg comparing
# AT vs MT for this model that contains 2 additional predictors
# besides am. So, we can say that the adjusted estimate for the
# expected change in mpg going from AT to MT is +2.94 gallons.
#
# Here are the residuals for the best multi-variable model (see Fig.4)
#
par(mfrow = c(2,2))
plot(model.best)
```




```r
#
# To further optimize this model, we examine mpg ~ wt + qsec controlled by am
#
model.bestv2 <- lm(mpg ~ factor(am):wt + factor(am):qsec, data=mtcars)
summary(model.bestv2)
```

```
##
## Call:
## lm(formula = mpg ~ factor(am):wt + factor(am):qsec, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.936 -1.402 -0.155  1.269  3.886
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           13.969      5.776    2.42   0.0226 *
## factor(am)AT:wt       -3.176      0.636   -4.99  3.1e-05 ***
## factor(am)MT:wt       -6.099      0.969   -6.30  9.7e-07 ***
## factor(am)AT:qsec      0.834      0.260    3.20   0.0035 **
## factor(am)MT:qsec      1.446      0.269    5.37  1.1e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.1 on 27 degrees of freedom
## Multiple R-squared:  0.895,  Adjusted R-squared:  0.879
## F-statistic: 57.3 on 4 and 27 DF,  p-value: 8.42e-13
```

```r
#
# With the revised best model we can see that it captured 89.5% of the total
# variance and the adjusted variance is 0.879 which is a bit better than
# what we obtained previously, 0.834. We note that wt is (lb/1000)
# abd qsec is    1/4 mile time. So, from the coefficients we see that
# when the car weight increased by 1000 lbs, the mpg decreased by
# -3.18 miles for AT cars and -6.09 for MT cars. For the qsec part,
# when the acceleration speed dropped and 1/4 mile time increased 1 secs,
# the mpg increased 0.834 miles for AT cars and 1.446 miles for MT cars.
# This implys that if the car has low acceleration at the same weight,
# MT cars are better for mpg. In summary, the mpg is largely determined
# by the interplay between weight, acceleration, and transmission type.

# Appendix

# One advantage of a linear model is that it can be used for predictions
# in addition to statistical testing. Here is what our best model predicts
# for each of the cars:
predict(model.bestv2)
```

```
##           Mazda RX4       Mazda RX4 Wag          Datsun 710
##               21.80               21.05               26.74
##      Hornet 4 Drive   Hornet Sportabout             Valiant
##               19.97               17.24               19.84
##          Duster 360           Merc 240D            Merc 230
##               15.84               20.51               23.06
##            Merc 280           Merc 280C          Merc 450SE
##               18.30               18.80               15.55
##          Merc 450SL         Merc 450SLC  Cadillac Fleetwood
##               16.80               16.97               12.29
## Lincoln Continental   Chrysler Imperial            Fiat 128
##               11.60               11.52               28.71
##         Honda Civic      Toyota Corolla       Toyota Corona
##               30.91               31.56               22.82
##    Dodge Challenger         AMC Javelin          Camaro Z28
##               16.86               17.48               14.62
##    Pontiac Firebird           Fiat X1-9       Porsche 914-2
##               15.97               29.50               25.07
##        Lotus Europa      Ford Pantera L        Ferrari Dino
##               29.18               15.61               19.49
##       Maserati Bora          Volvo 142E
##               13.31               23.92
```

```r
#
# What if we went back to a single variable model that depends on wt only
#
modelw <- lm(mpg ~ wt, data=mtcars)
summary(modelw)
```

```
##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
```

```
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.543 -2.365 -0.125  1.410  6.873
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   37.285      1.878   19.86  < 2e-16 ***
## wt            -5.344      0.559   -9.56  1.3e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.05 on 30 degrees of freedom
## Multiple R-squared:  0.753,  Adjusted R-squared:  0.745
## F-statistic: 91.4 on 1 and 30 DF,  p-value: 1.29e-10
```

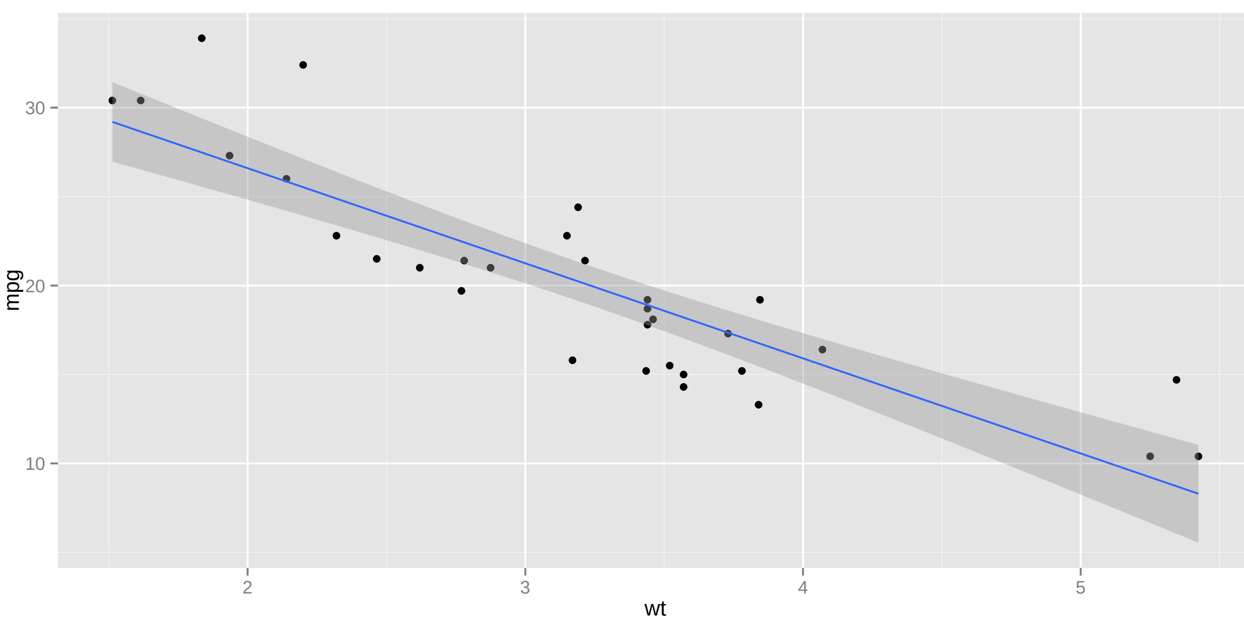
```
#
# Here, we have an eqn of mpg = 37.285 - 5.3445*wt
# So, if we had a car of 4500 lbs (4.5 * wt)
37.285 - 5.3445*4.5
```

```
## [1] 13.23
```

```
#which predicts a gas mileage of 13.2 mpg. The shortcut for doing
# this involves creating a data frame and using that with the predict function.
newcar=data.frame(wt=4.5)
predict(modelw,newcar)
```

```
##     1
## 13.24
```

```
# You can visualize the dependence of the mpg on wt using the geom_smooth
# method built into ggplot.
library(ggplot2)
ggplot(mtcars, aes(wt,mpg)) + geom_point() + geom_smooth(method="lm")
```



```r
# Here the gray area is the uncertainty in the fit or its 95% confidence interval
# of where the true trend line could be.
# Now lets add in the number of cylinders(cyl) and displacement (disp)
# and look at the trend.
ggplot(mtcars, aes(x=wt, y=mpg, col=cyl, size=disp)) + geom_point()
```