Project

Winston

November 9, 2014

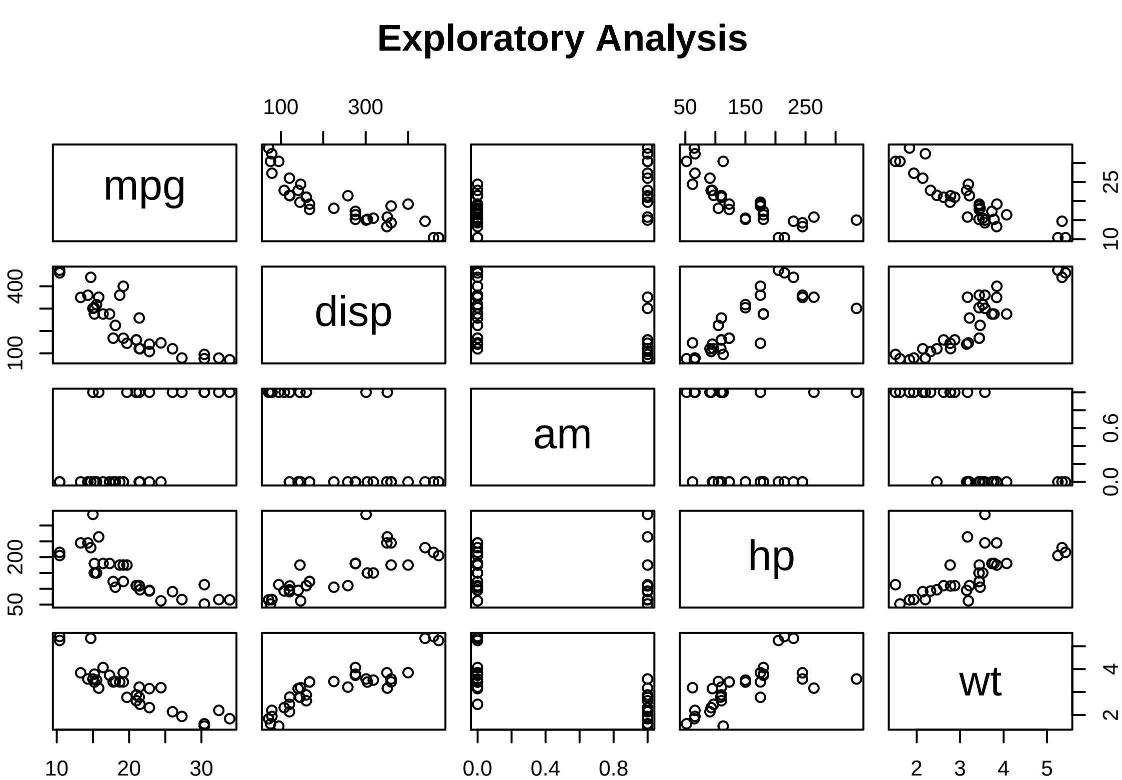
Executive Summary

This analysis looks at the energy efficiency (~ mpg) of cars in the Motor Trend data set and the factors affecting it. Particularly, it addresses the hyptohesis that manual transmissions are more efficient (have higher mpg) than automatic transmissions.

To address this, the analysis takes the approach of linearizing the "physics" by looking at the affects of confounding variables like horsepower (hp) and vehicle weight (wt) on gallons per mile.

exploratory data analysis

```
data(mtcars)
library(plyr)

mtcars2=mutate(mtcars, gpm=1/mpg)

pairs(mpg~disp+am+hp+wt,data=mtcars,
   main="Exploratory Analysis")
```



There are data for 32 different cars in the data set.
The data present are *mpg, cyl, disp, hp, drat, wt, qsec, vs, am, gear, carb.*

mpg Miles/(US) gallon cyl Number of cylinders disp Displacement (cu.in.) hp Gross horsepower drat Rear axle ratio wt Weight (lb/1000) qsec 1/4 mile time vs V/S am Transmission (0 = automatic, 1 = manual) gear Number of forward gears carb Number of carburetors

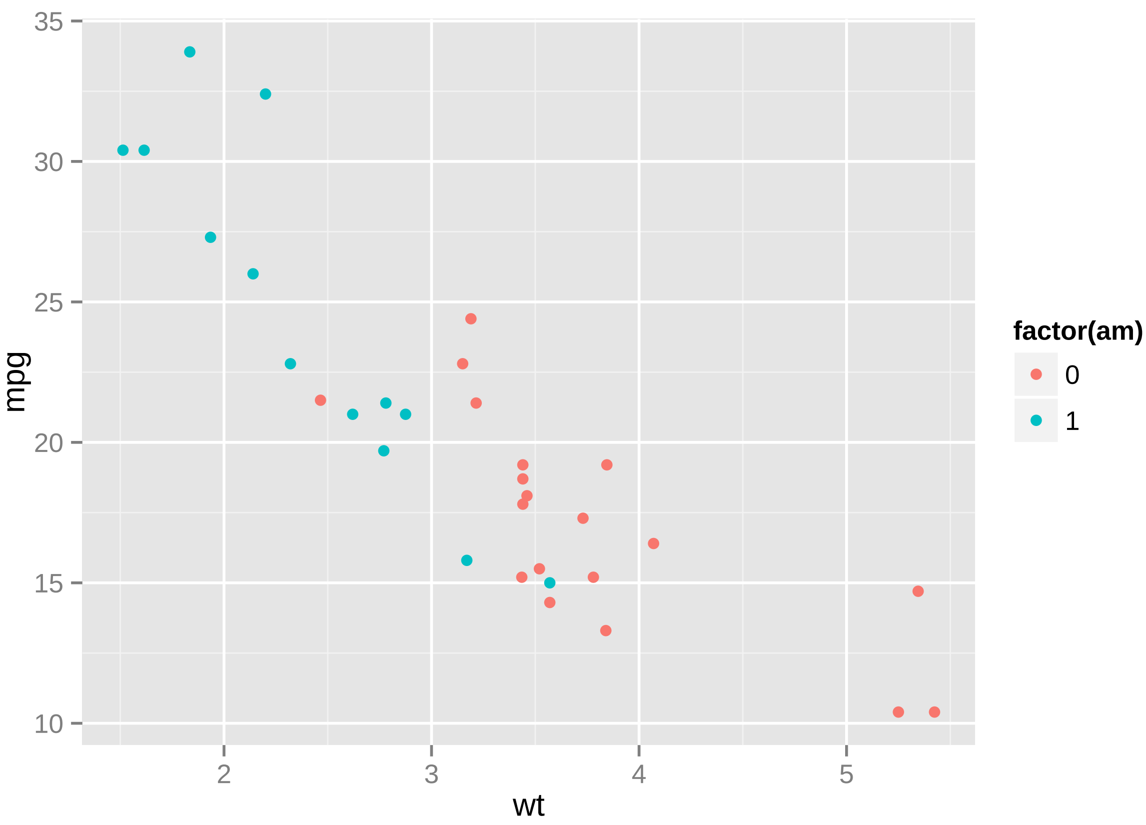
The exploratory analysis shows there are multiple dependencies. To understand how transmission type affects mpg we need account for these other factors.

Note that

As you can see from the graph, displacement and horsepower tend to follow weigth. So we'll focus on correcting for the weight factor in mpg and look at residual.

linear Model

```
library(plyr)
library(ggplot2)

p<-ggplot(mtcars, aes(wt, mpg, color=factor(am))) + geom_point()
p
```




```
f1<-lm(mpg~ wt, data=mtcars)
rf1<-residuals(f1)

rf1
```

```
##           Mazda RX4      Mazda RX4 Wag          Datsun 710
```

```
##          -2.2826106          -0.9197704          -2.0859521
##      Hornet 4 Drive   Hornet Sportabout             Valiant
##           1.2973499          -0.2001440          -0.6932545
##          Duster 360           Merc 240D            Merc 230
##          -3.9053627           4.1637381           2.3499593
##            Merc 280           Merc 280C          Merc 450SE
##           0.2998560          -1.1001440           0.8668731
##          Merc 450SL         Merc 450SLC  Cadillac Fleetwood
##          -0.0502472          -1.8830236           1.1733496
## Lincoln Continental   Chrysler Imperial            Fiat 128
##           2.1032876           5.9810744           6.8727113
##         Honda Civic      Toyota Corolla       Toyota Corona
##           1.7461954           6.4219792          -2.6110037
##    Dodge Challenger         AMC Javelin          Camaro Z28
##          -2.9725862          -3.7268663          -3.4623553
##    Pontiac Firebird           Fiat X1-9       Porsche 914-2
##           2.4643670           0.3564263           0.1520430
##        Lotus Europa      Ford Pantera L        Ferrari Dino
##           1.2010593          -4.5431513          -2.7809399
##       Maserati Bora          Volvo 142E
##          -3.2053627          -1.0274952
```

```
#p<-ggplot(mtcars, aes(wt, rf1, color=factor(am))) + geom_point()
```

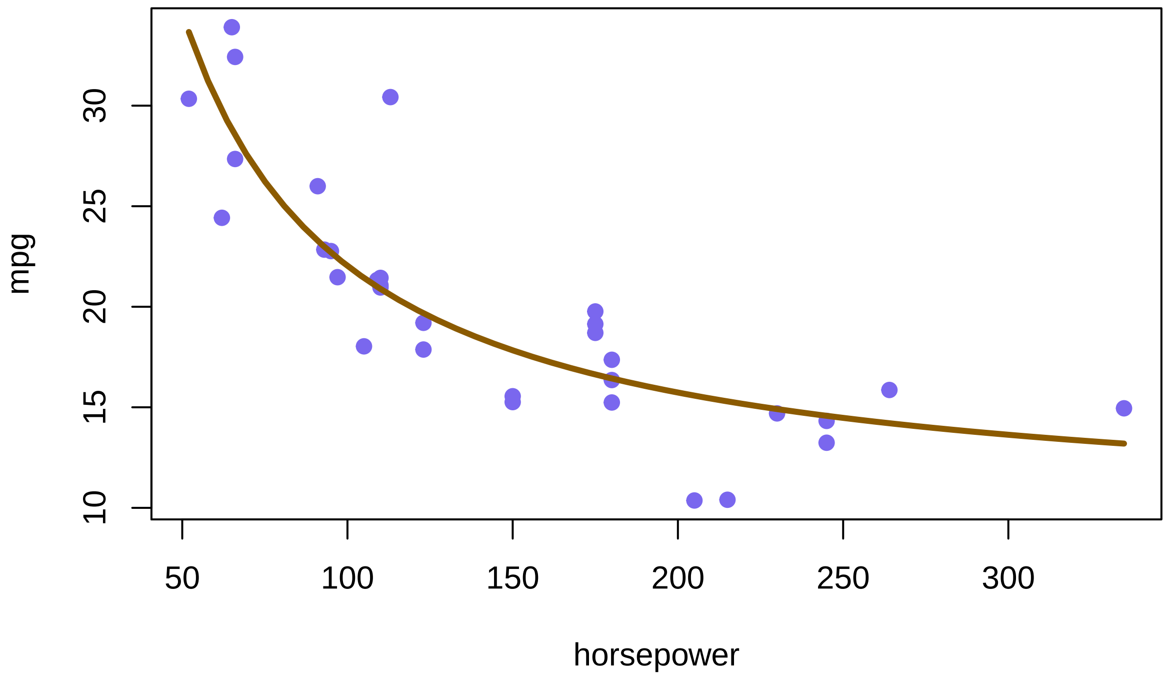
nonlinear model of mpg versus wt

To fit the data try a functional relationship *mpg~ a + b/wt* with a and b constants to be determined. The graph below shows the fit.

```
carnls <- nls(mpg~a1 + b1/hp, mtcars, start = list(a1=1, b1=50))


carnls
```

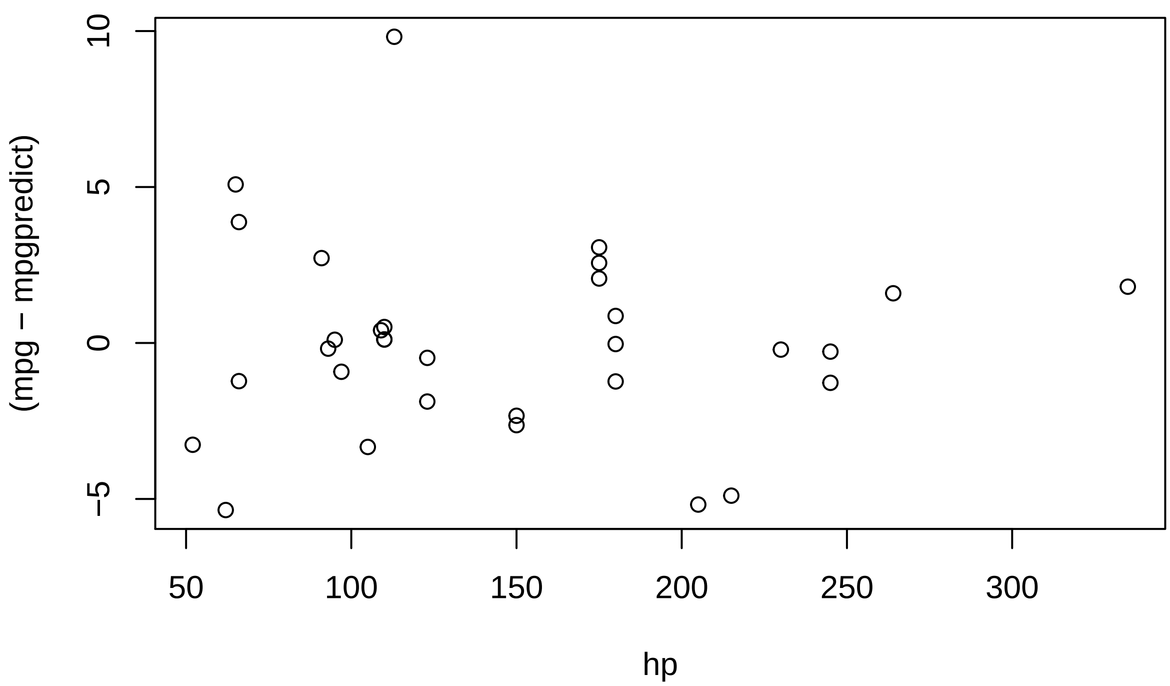
```
## Nonlinear regression model
##   model: mpg ~ a1 + b1/hp
##    data: mtcars
##       a1       b1
##    9.434 1259.881
##  residual sum-of-squares: 294.9
##
## Number of iterations to convergence: 1
## Achieved convergence tolerance: 3.841e-08
```

```
ss<-seq(from = min(mtcars$hp), to = max(mtcars$hp), length = 50)


plot(jitter(mpg,4)~hp, data=mtcars, pch=19, col="slateblue2", xlab="horsepower", ylab="mpg")


yy <- predict(carnls,list(hp = ss))

lines(ss, yy, col = "orange4", lwd=3)
```

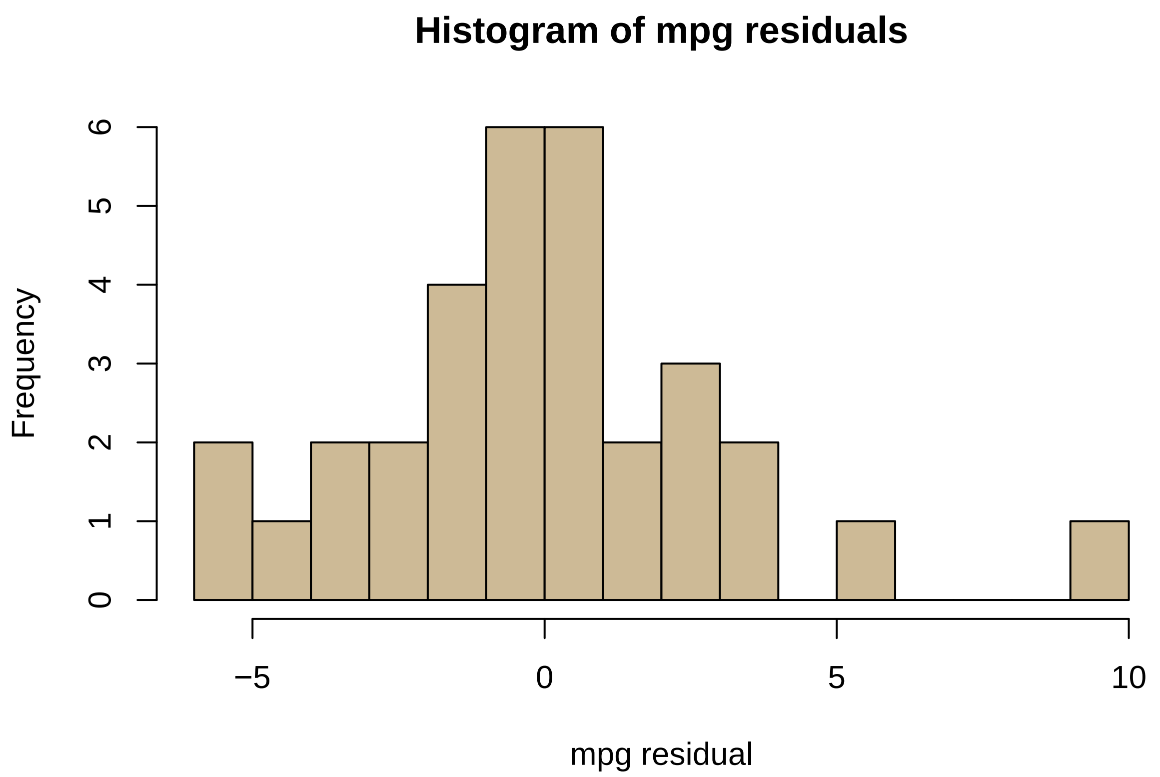


Model Residuals

```
mpgpredict<-predict(carnls)

plot((mpg-mpgpredict)~hp, data=mtcars)
```



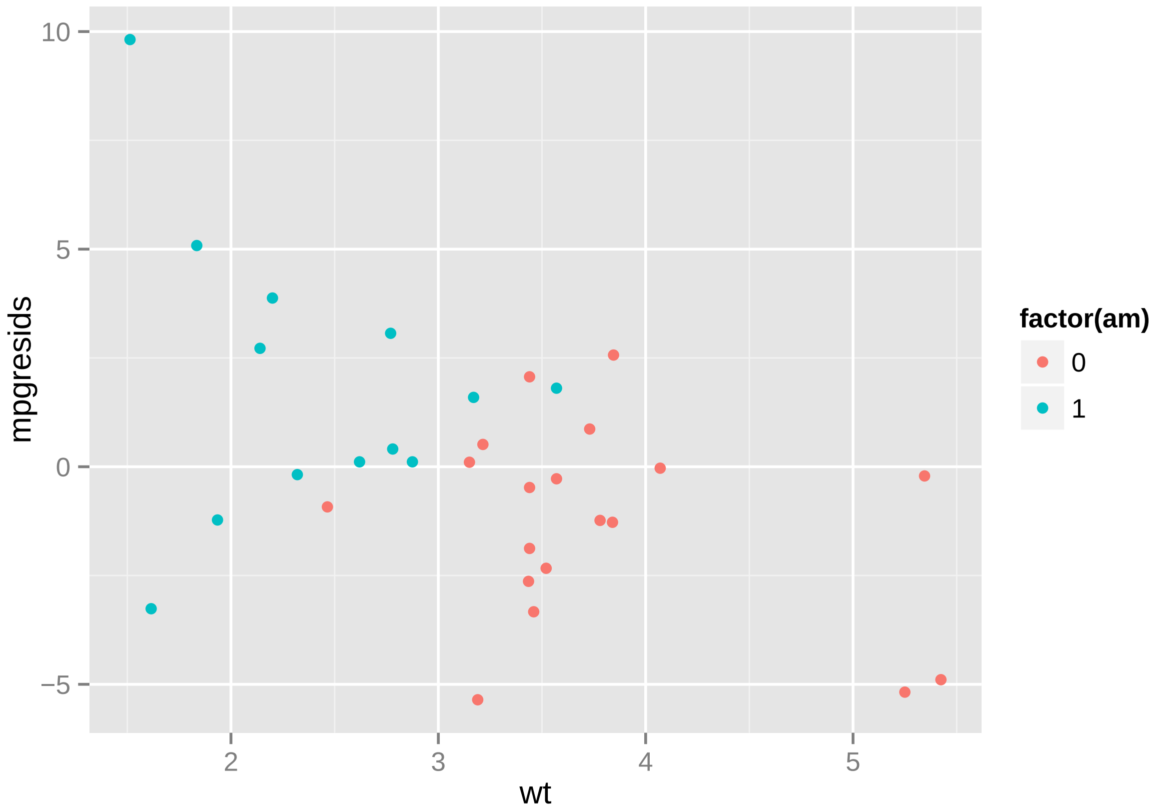
```r
hist(mtcars$mpg-mpgpredict, breaks=18, col="wheat3", xlab="mpg residual", main="Histogram of mpg residu
```



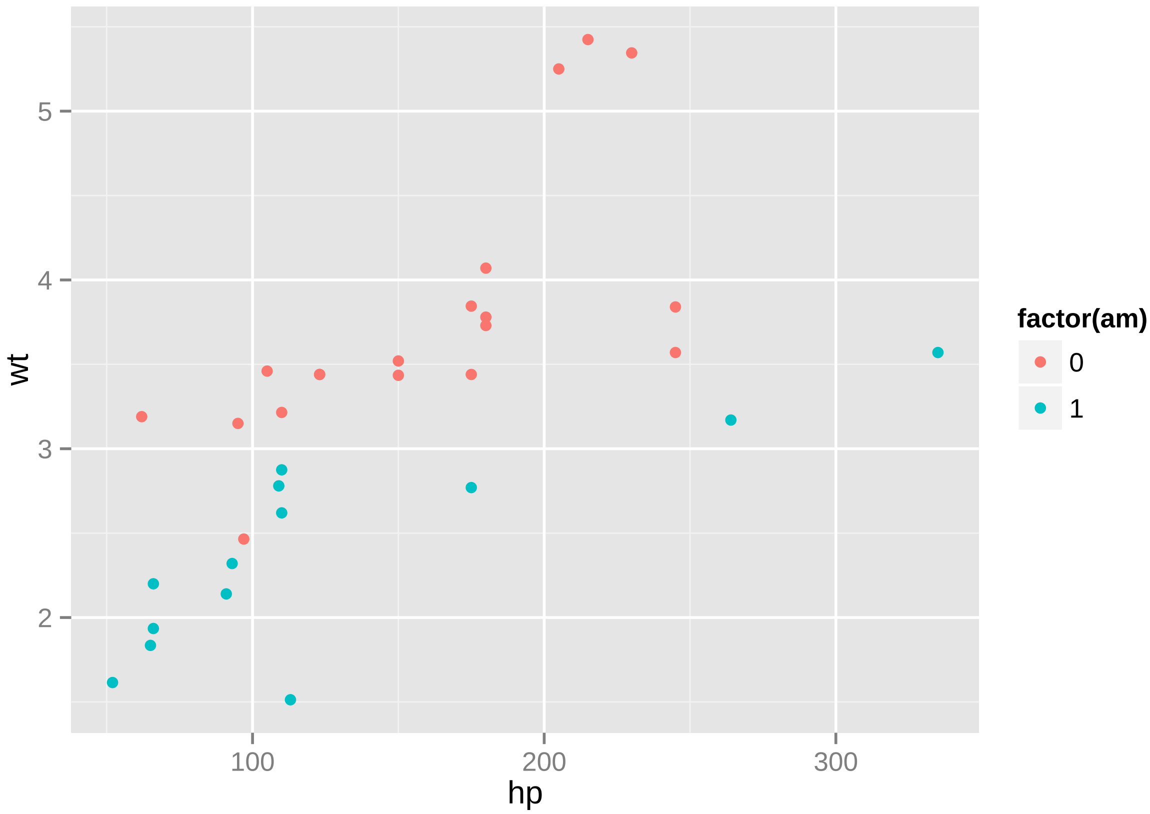

```r
residuals(carnls)
```

```
##  [1]  0.1121984  0.1121984 -0.1814456  0.5121984  2.0663423 -3.3332046
##  [7] -0.2767090 -5.3549997  0.1037567 -0.4772728 -1.8772728 -0.0336766
## [13]  0.8663234 -1.2336766 -5.1800986 -4.8942493 -0.2120811  3.8765556
## [19] -3.2628193  5.0828771 -0.9228019 -2.3335444 -2.6335444 -1.2767090
## [25]  2.5663423 -1.2234444  2.7208158  9.8162727  1.5933859  3.0663423
## [31]  1.8048236  0.4071208
## attr(,"label")
## [1] "Residuals"
```

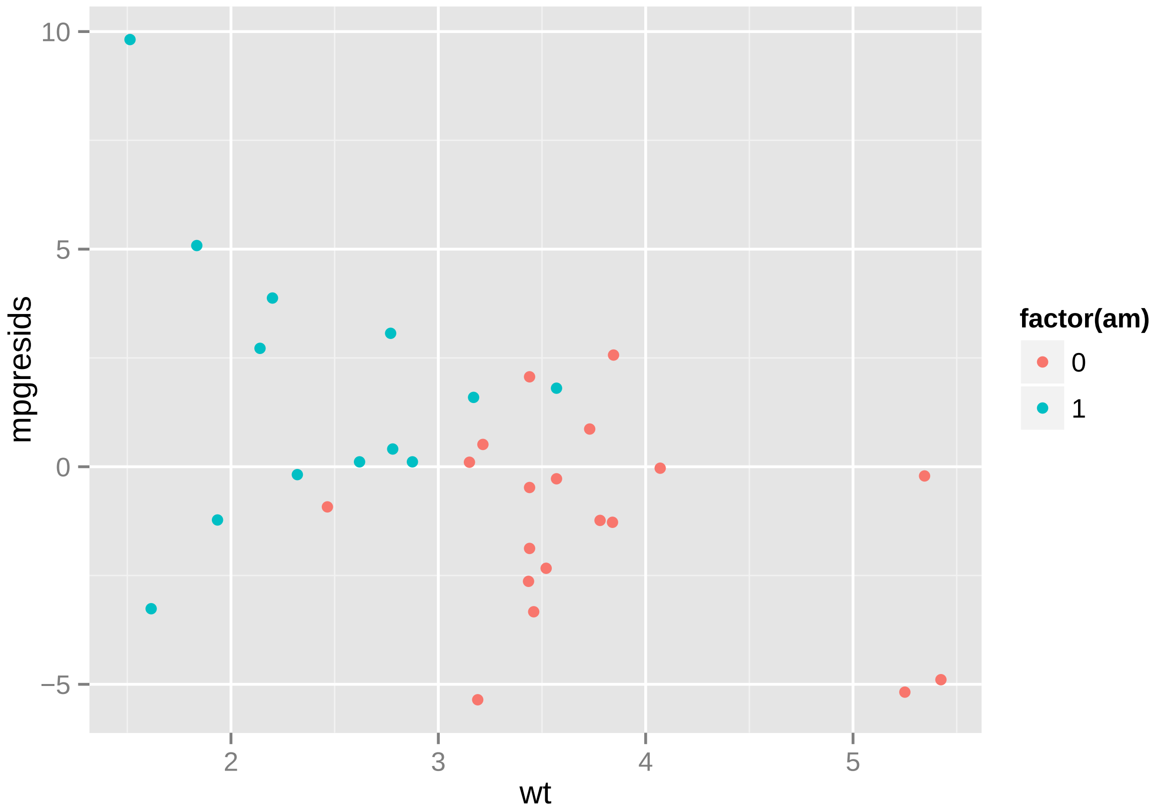
```r
library(plyr)

mpgresids <- mtcars$mpg-mpgpredict

mtcars2<-cbind(mtcars, mpgresids)

p<-ggplot(mtcars2, aes(wt, mpgresids, color=factor(am))) + geom_point()

p
```



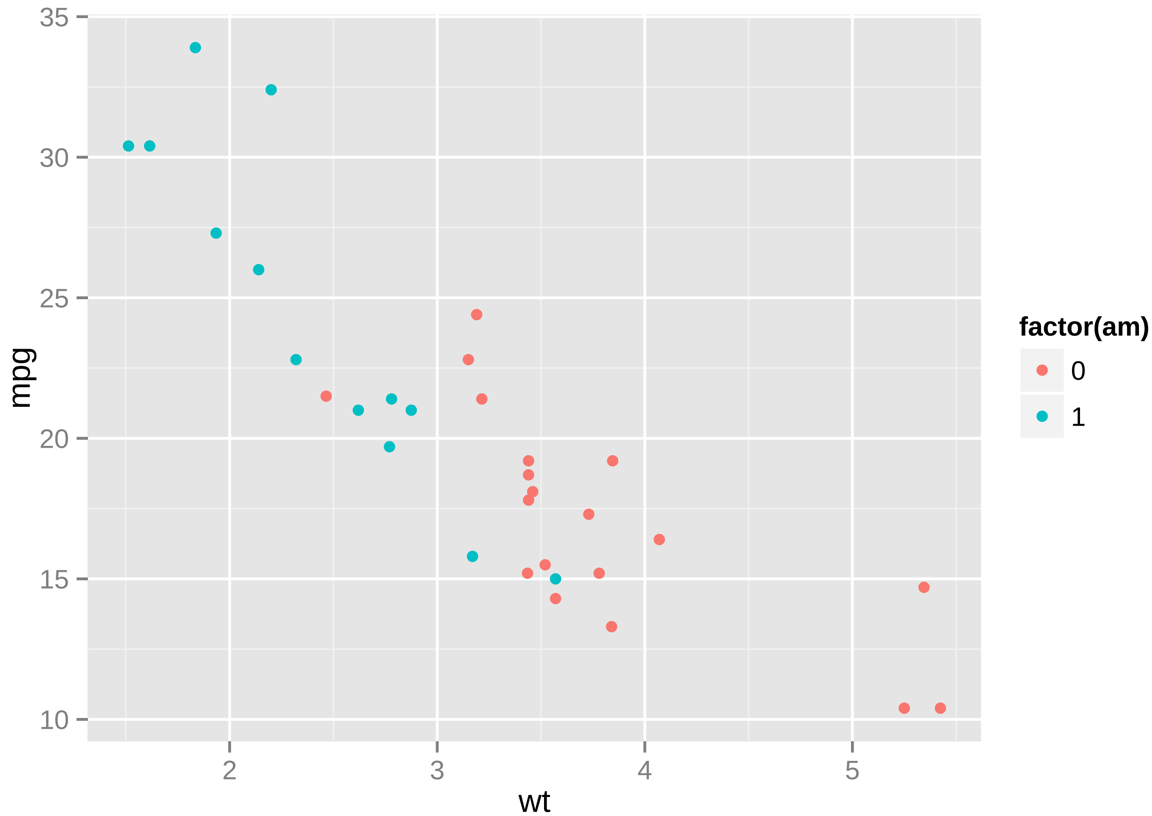

```
p<-ggplot(mtcars, aes(hp, wt, color=factor(am))) + geom_point()

p
```




```
library(plyr)

## calculate residuals ad bind to mtcars to do some plotting

mpgresids <- mtcars$mpg-mpgpredict

mtcars2<-cbind(mtcars, mpgresids)

p<-ggplot(mtcars2, aes(wt, mpgresids, color=factor(am))) + geom_point()

p
```




```
p<-ggplot(mtcars, aes(wt, mpg, color=factor(am))) + geom_point()

p
```




```
##here we do statistics on the residuals


autoresids <- mtcars2[mtcars$am==0,]
manresids <- mtcars2[mtcars$am==1,]

meanautoresids<-mean(autoresids$mpgresids)
sdautoresids<-sd(autoresids$mpgresids)
nautoresids <- nrow(autoresids)

meanmanresids<-mean(manresids$mpgresids)
sdmanresids<-sd(manresids$mpgresids)
nmanresids <- nrow(manresids)


aaa<-t.test(autoresids$mpgresids, manresids$mpgresids, alternative="less")
pv<-aaa$p.value
lc<-aaa$conf.int[1]
uc<-aaa$conf.int[2]
```