Ankit_Nikhil_Saket_Utkarsh_ASSG2

October 11, 2023

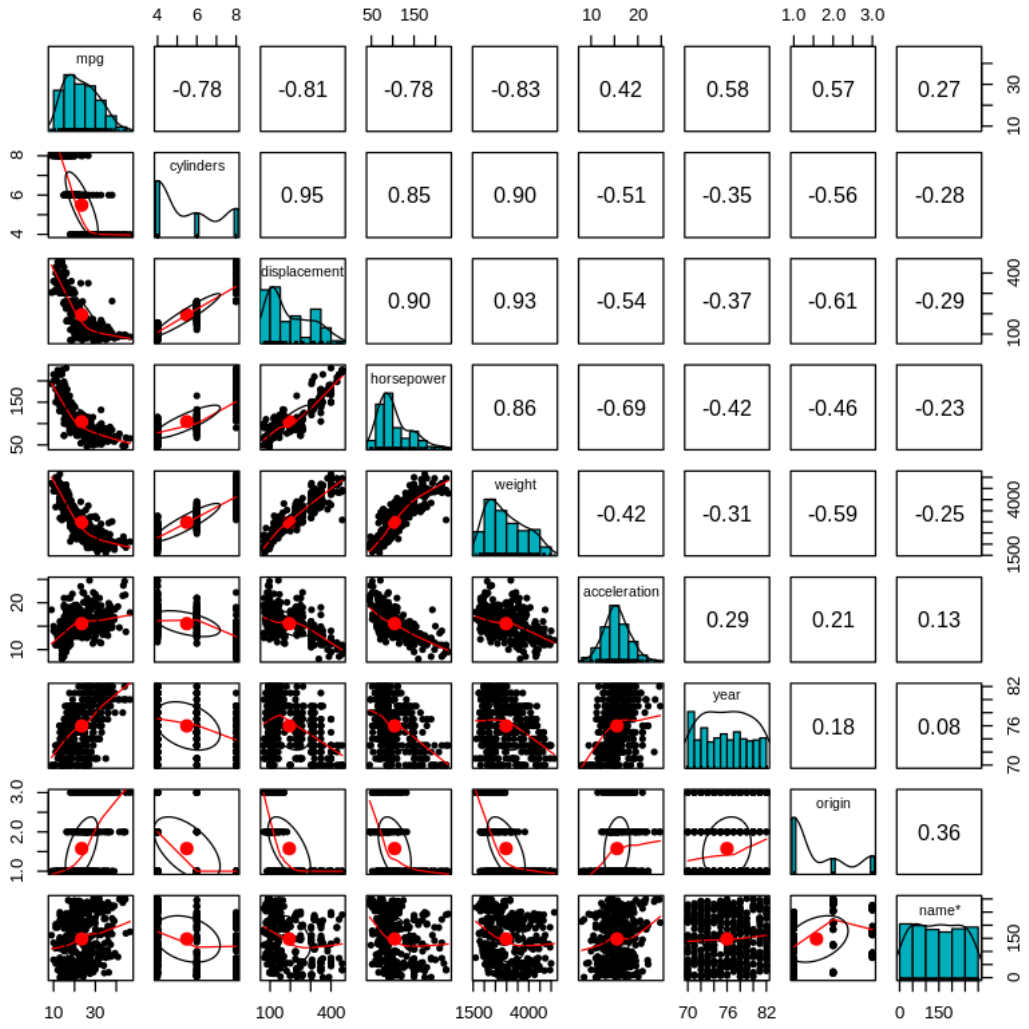
[29]:
```
df = read.csv('Auto.csv')
```

[30]:
```
table(df$cylinders)
```

```
  3   4   5   6   8
  4 199   3  83 103
```

[31]:
```
# To make our lives easier we have clubbed 3 cylinder and 5 cylinder with 4 and␣
 ↪6 respectively
df$cylinders[df$cylinders == 3] <- 4

df$cylinders[df$cylinders == 5] <- 6
```

[32]:
```
pairs.panels(df,
             method = "pearson",
             hist.col = "#00AFBB",
             density = TRUE,
             ellipses = TRUE
             )
```

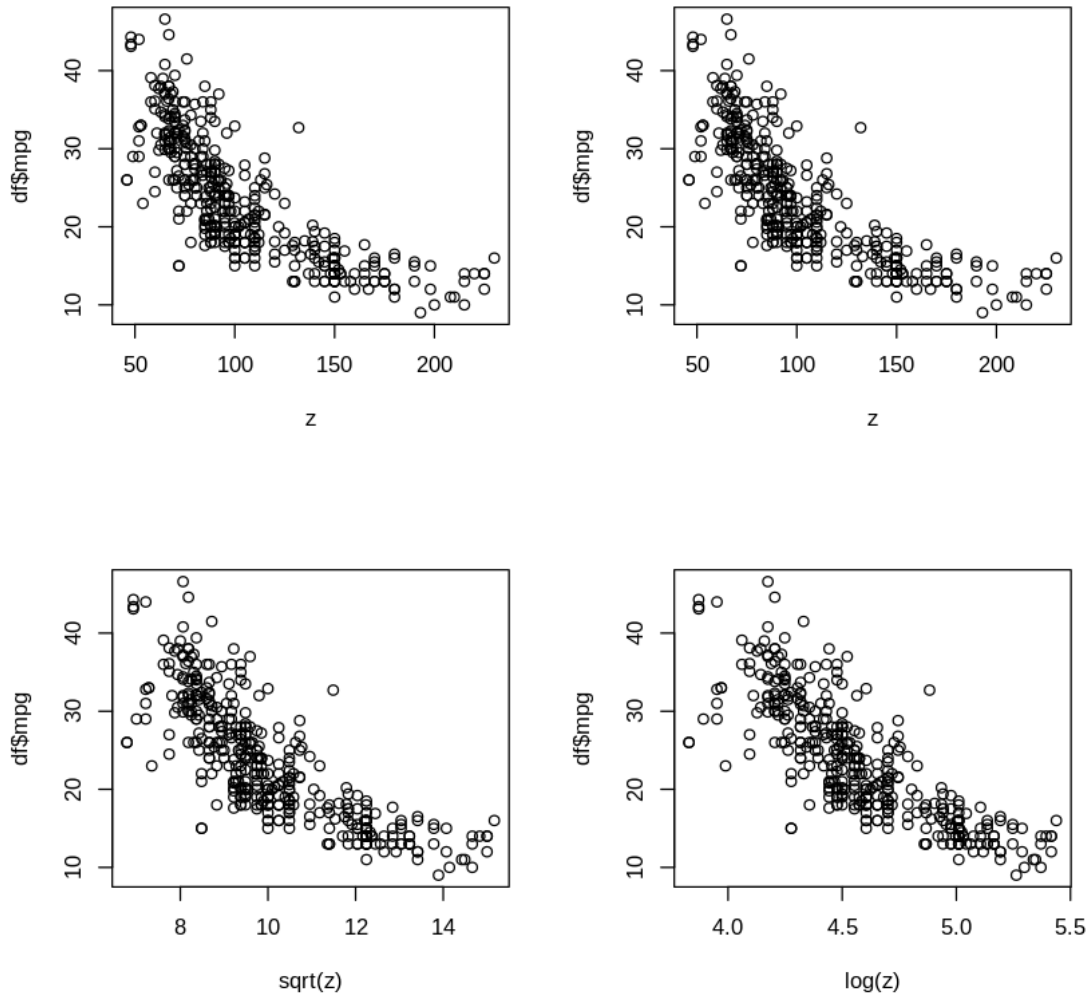


1 Nonlinearity

[33]:
```
z = df$horsepower
par(mfrow = c(2,2))
plot( df$mpg ~ z )
cat('Simple Correlation',cor( df$mpg ,z ),'\n' )

plot( df$mpg ~ z**2 )
cat('Correlation with square term',cor( df$mpg ,z**2 ),'\n' )

plot( df$mpg ~ sqrt(z) )
cat('Correlation with sqrt term',cor( df$mpg ,sqrt(z) ),'\n' )
```

```
plot( df$mpg ~ log(z) )
cat('Correlation with log term',cor( df$mpg ,log(z)) )

#log preferred
```

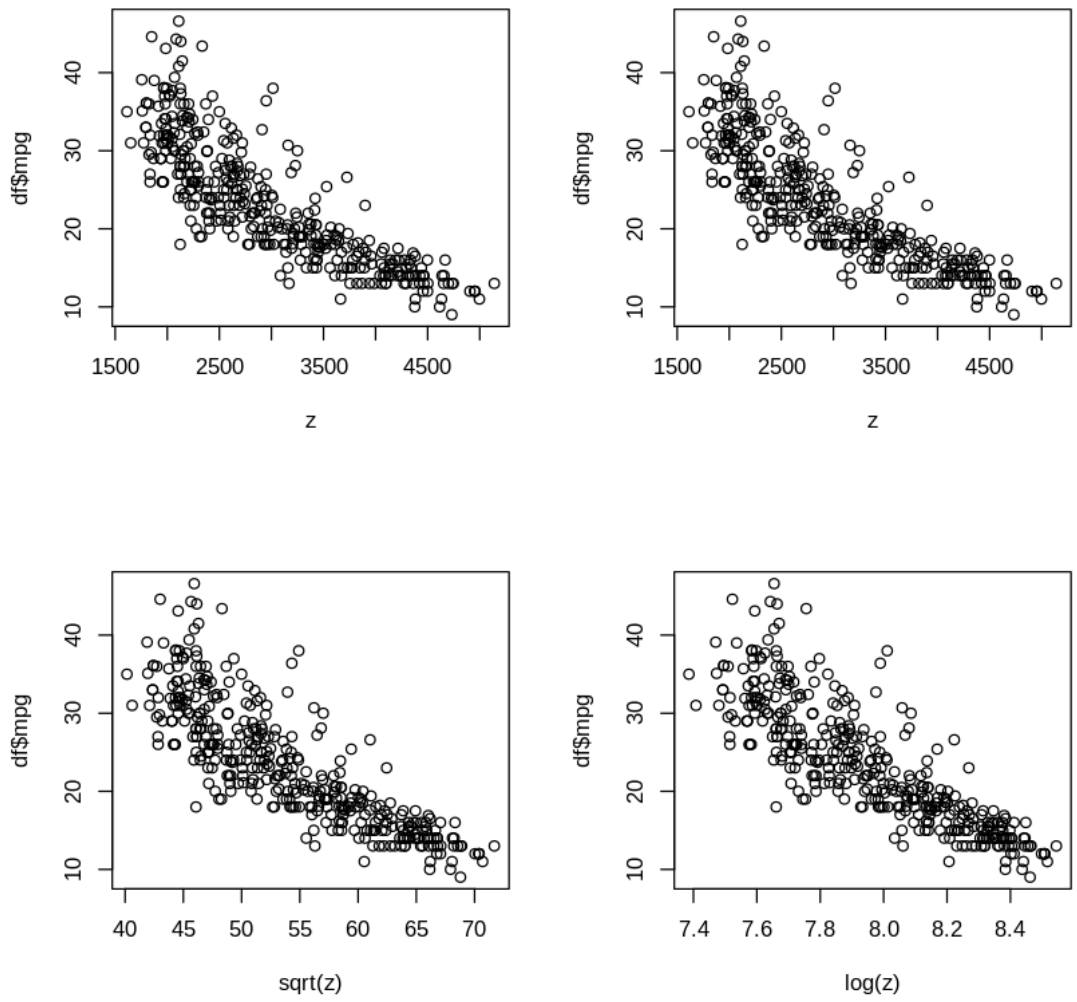
```
Simple Correlation -0.7784268
Correlation with square term -0.712297
Correlation with sqrt term -0.8023114
Correlation with log term -0.8175174
```



[34]:
```
z = df$weight
par(mfrow = c(2,2))
```

```
plot( df$mpg ~ z )
cat('Simple Correlation',cor( df$mpg ,z ),'\n' )

plot( df$mpg ~ z**2 )
cat('Correlation with square term',cor( df$mpg ,z**2 ),'\n' )

plot( df$mpg ~ sqrt(z) )
cat('Correlation with sqrt term',cor( df$mpg ,sqrt(z) ),'\n' )

plot( df$mpg ~ log(z) )
cat('Correlation with log term',cor( df$mpg ,log(z)) )

# Linear works
```

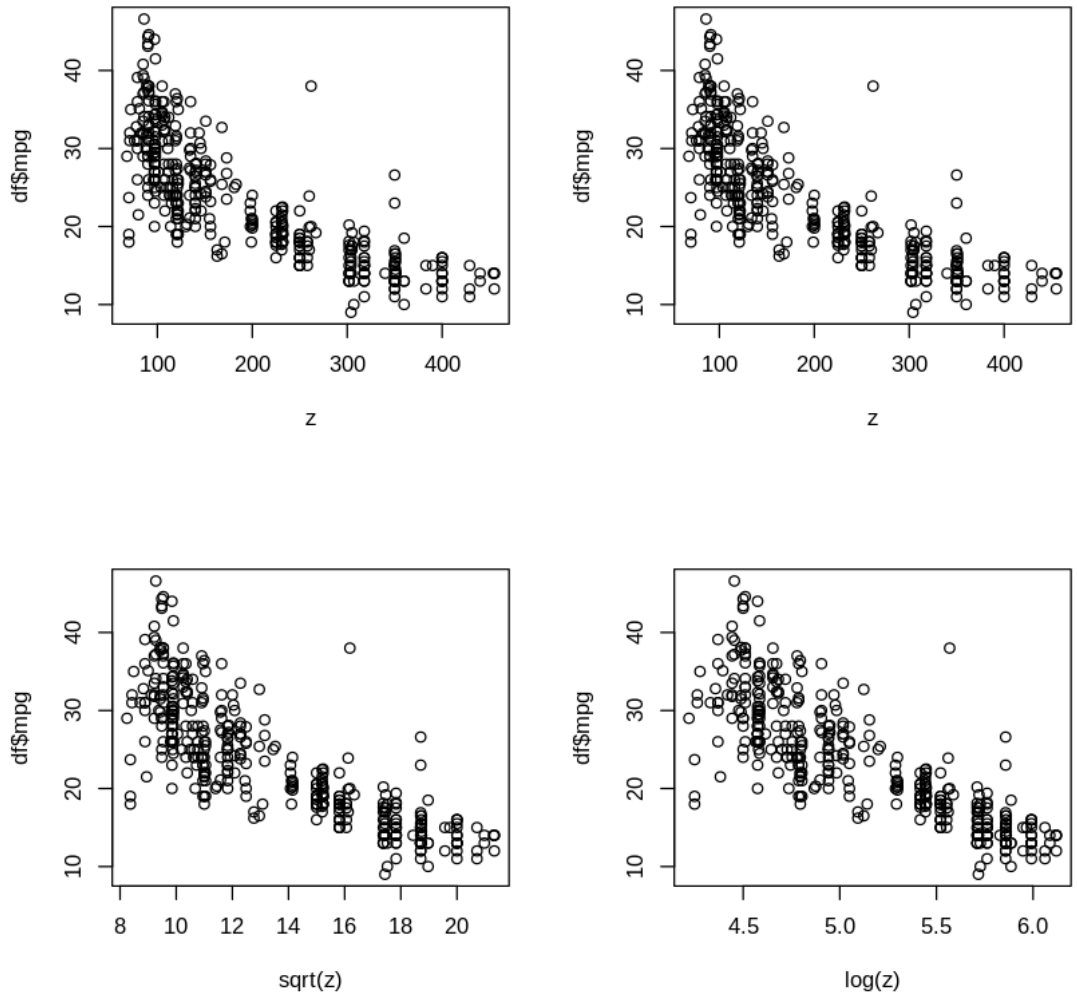
```
Simple Correlation -0.8322442
Correlation with square term -0.8066816
Correlation with sqrt term -0.8400951
Correlation with log term -0.8441938
```



[35]:
```
z = df$displacement
par(mfrow = c(2,2))
plot( df$mpg ~ z )
cat('Simple Correlation',cor( df$mpg ,z ),'\n' )

plot( df$mpg ~ z**2 )
cat('Correlation with square term',cor( df$mpg ,z**2 ),'\n' )

plot( df$mpg ~ sqrt(z) )
cat('Correlation with sqrt term',cor( df$mpg ,sqrt(z) ),'\n' )

plot( df$mpg ~ log(z) )
cat('Correlation with log term',cor( df$mpg ,log(z)) )
```

```
#log preferred
```

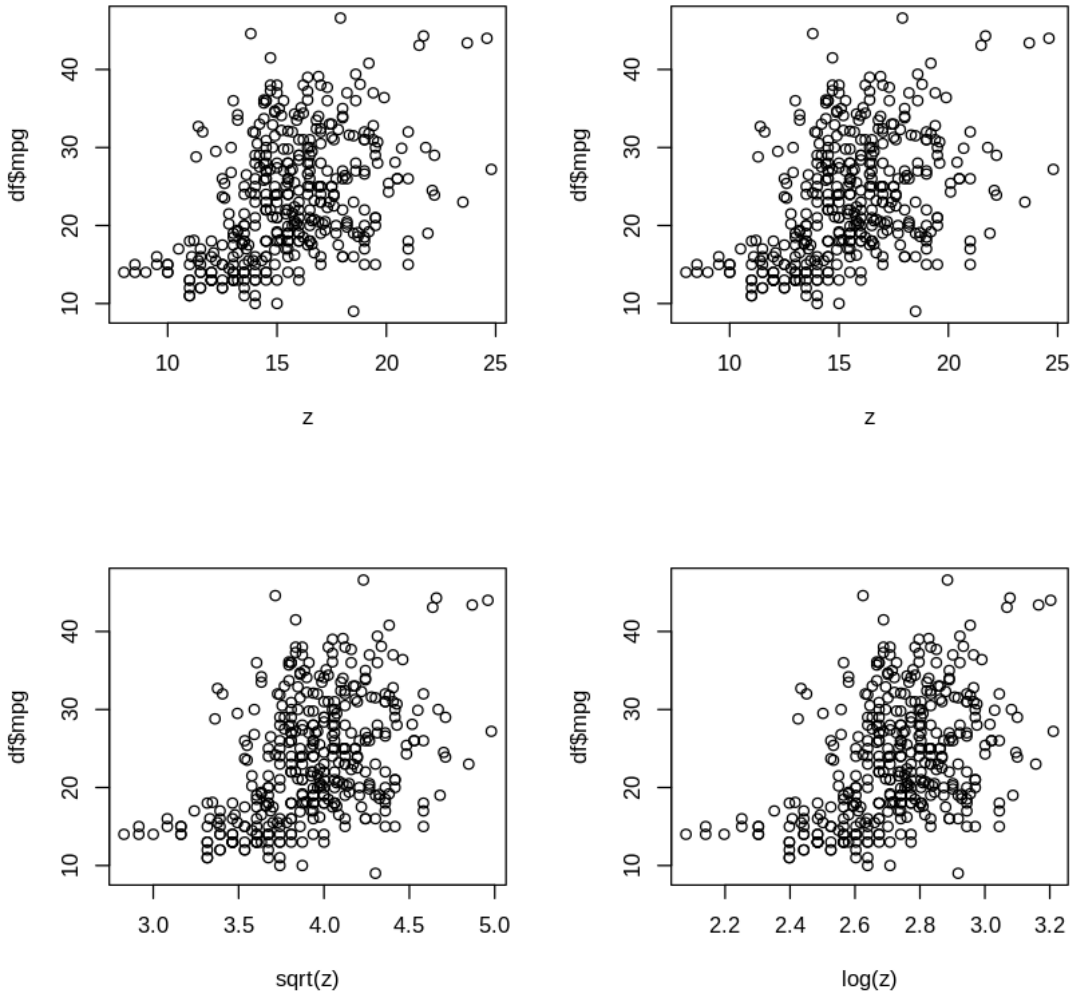
```
Simple Correlation -0.8051269
Correlation with square term -0.7523545
Correlation with sqrt term -0.8213314
Correlation with log term -0.8284533
```



[36]:
```
z = df$acceleration
par(mfrow = c(2,2))
plot( df$mpg ~ z )
cat('Simple Correlation',cor( df$mpg ,z ),'\n' )
```

```
plot( df$mpg ~ z**2 )
cat('Correlation with square term',cor( df$mpg ,z**2 ),'\n' )

plot( df$mpg ~ sqrt(z) )
cat('Correlation with sqrt term',cor( df$mpg ,sqrt(z) ),'\n' )

plot( df$mpg ~ log(z) )
cat('Correlation with log term',cor( df$mpg ,log(z)) )

#linear works
```

```
Simple Correlation 0.4233285
Correlation with square term 0.4037617
Correlation with sqrt term 0.4306775
Correlation with log term 0.4359007
```

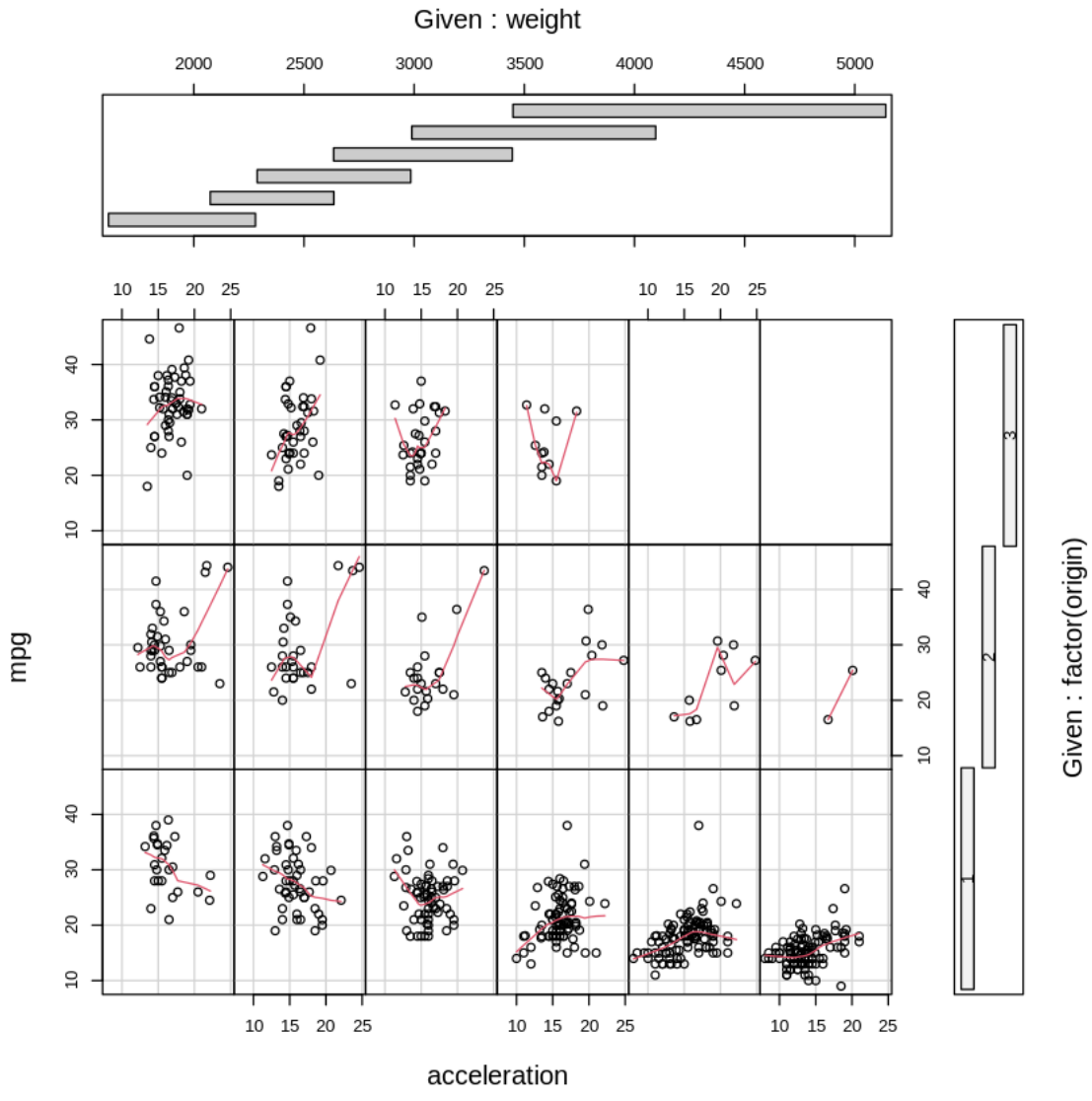


2 Interactions

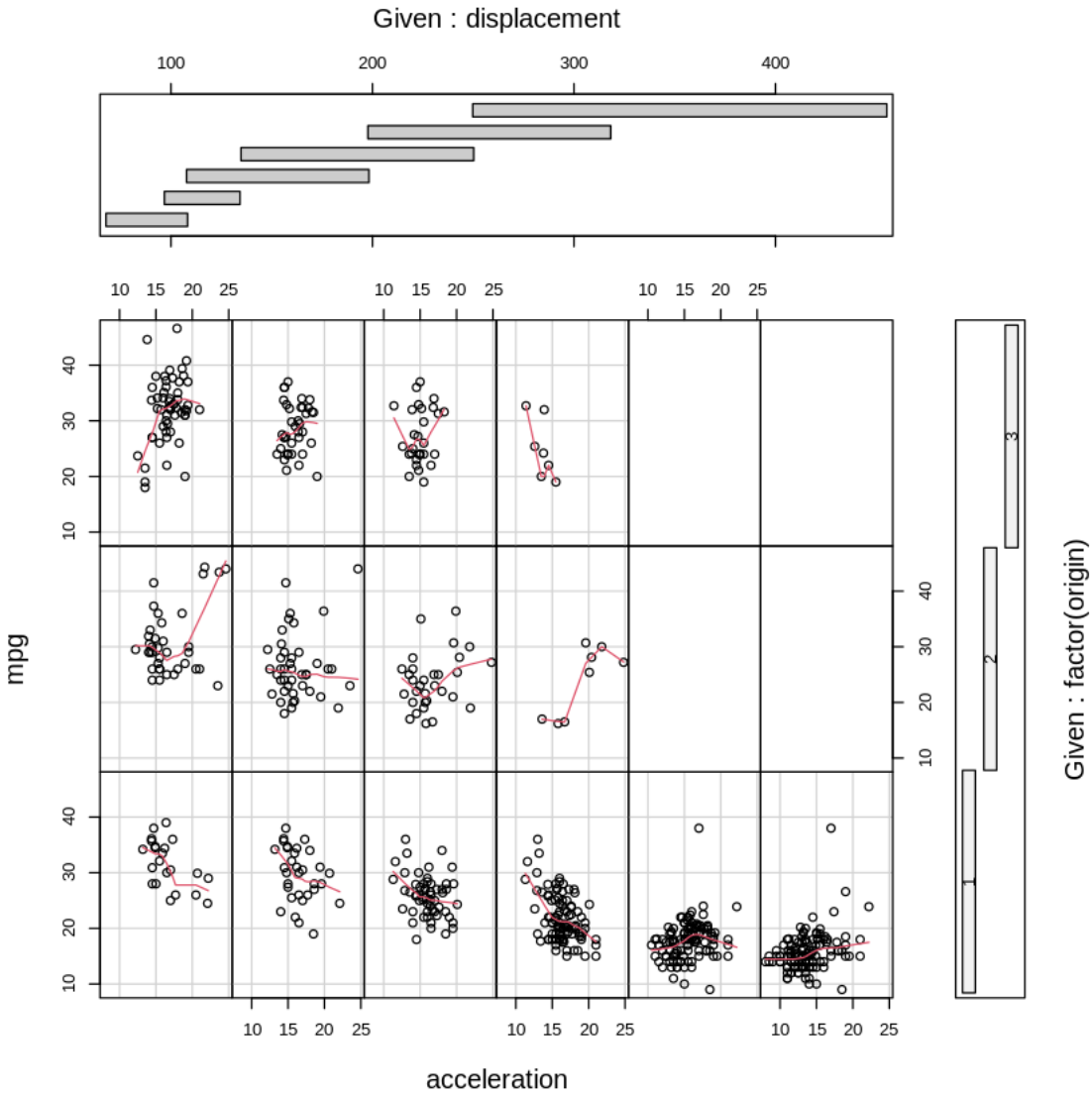
The coplot() function does it's best to split the data up to ensure there are an adequate number of data points in each panel

3 Acceleration

[37]:
```
coplot(mpg ~ acceleration | weight*factor(origin),
        number = 6, rows = 1,
         panel= panel.smooth,data = df)

# Interaction effect exists
```

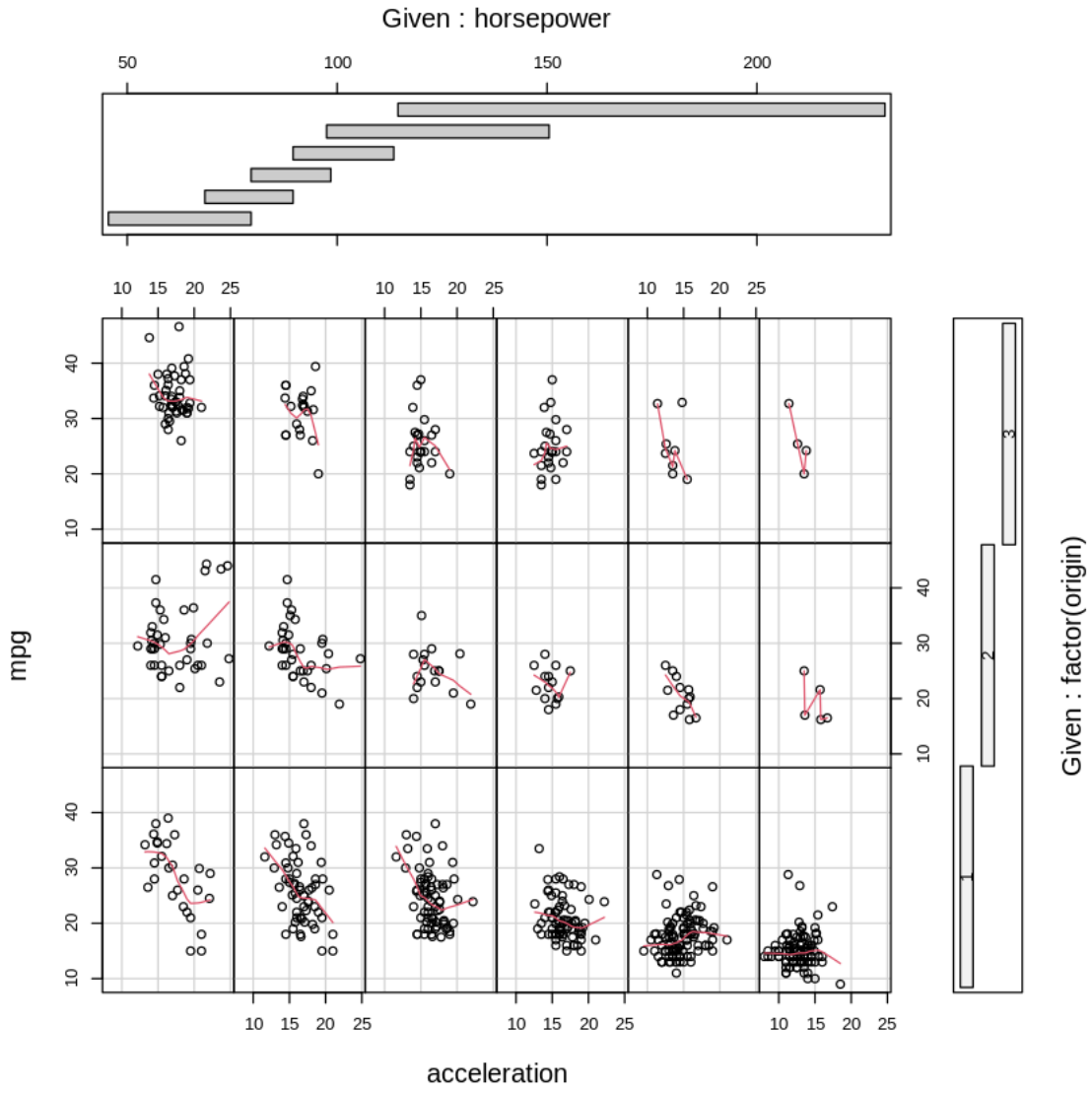


[38]:
```
coplot(mpg ~ acceleration | displacement*factor(origin),
       number = 6, rows = 1,
        panel= panel.smooth,data = df)

# Interaction effect exists
```

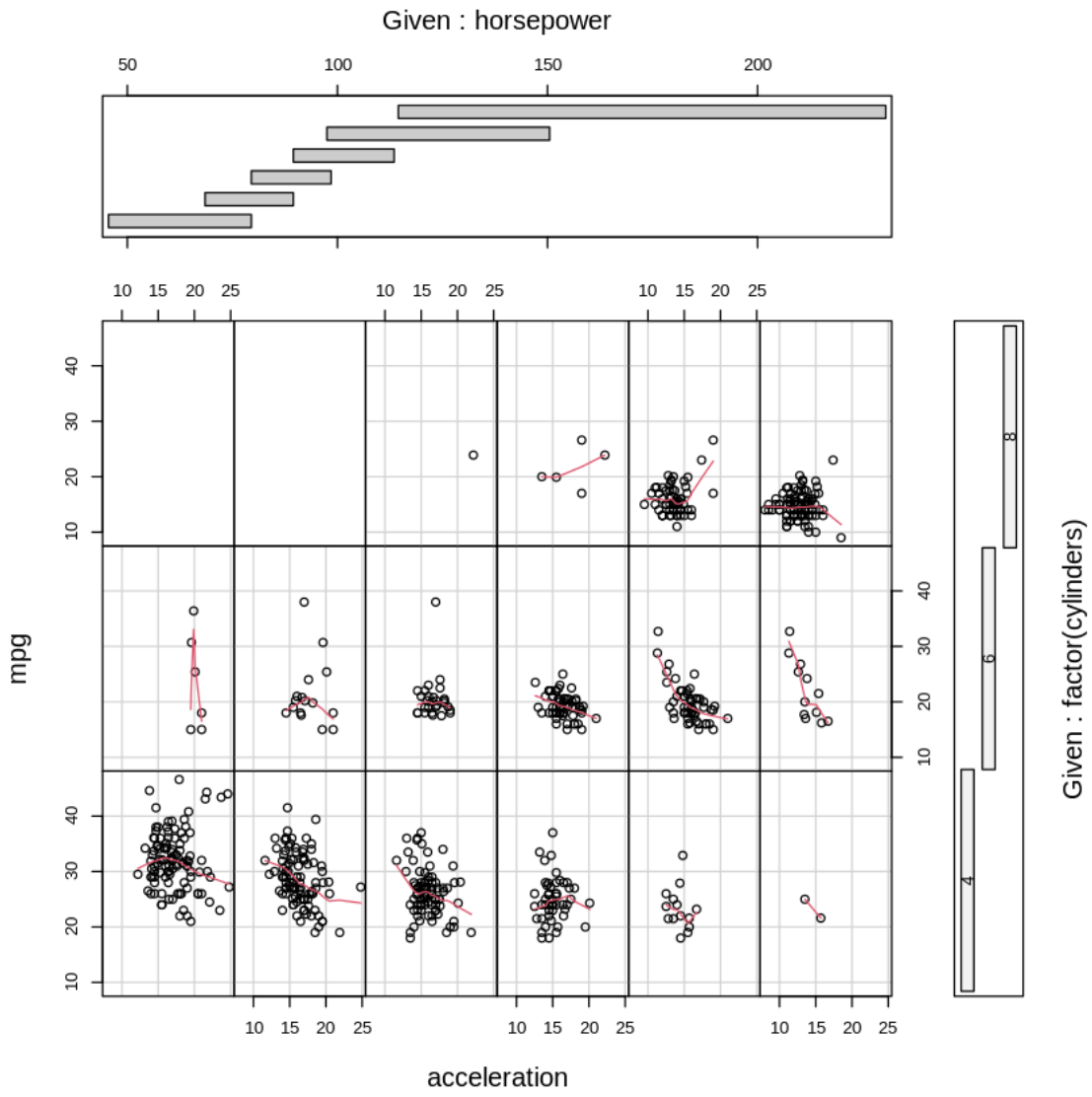


[39]:
```
coplot(mpg ~ acceleration | horsepower*factor(origin),
       number = 6, rows = 1,
        panel= panel.smooth,data = df)
```

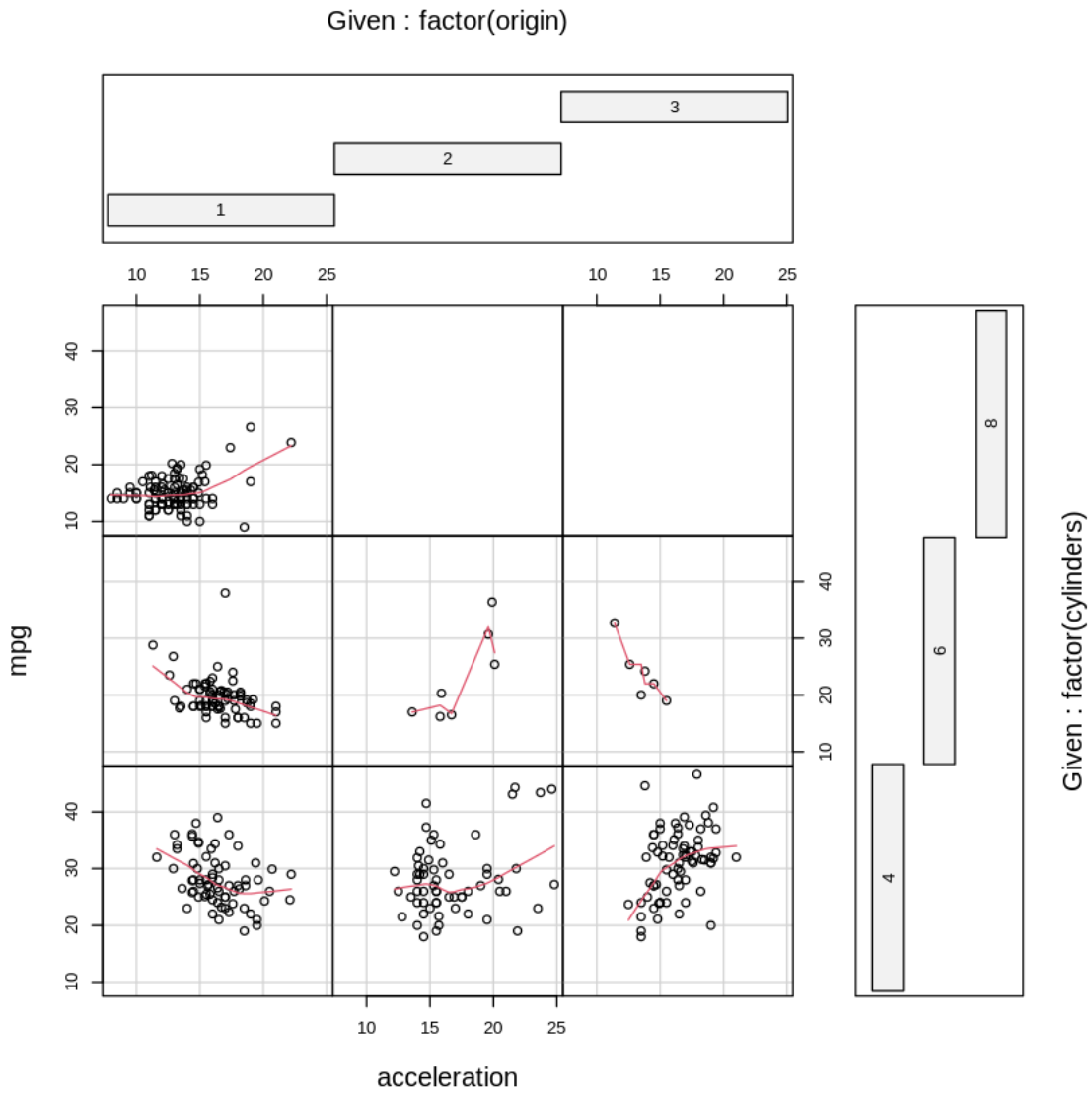
```
# Interaction effect exists
```



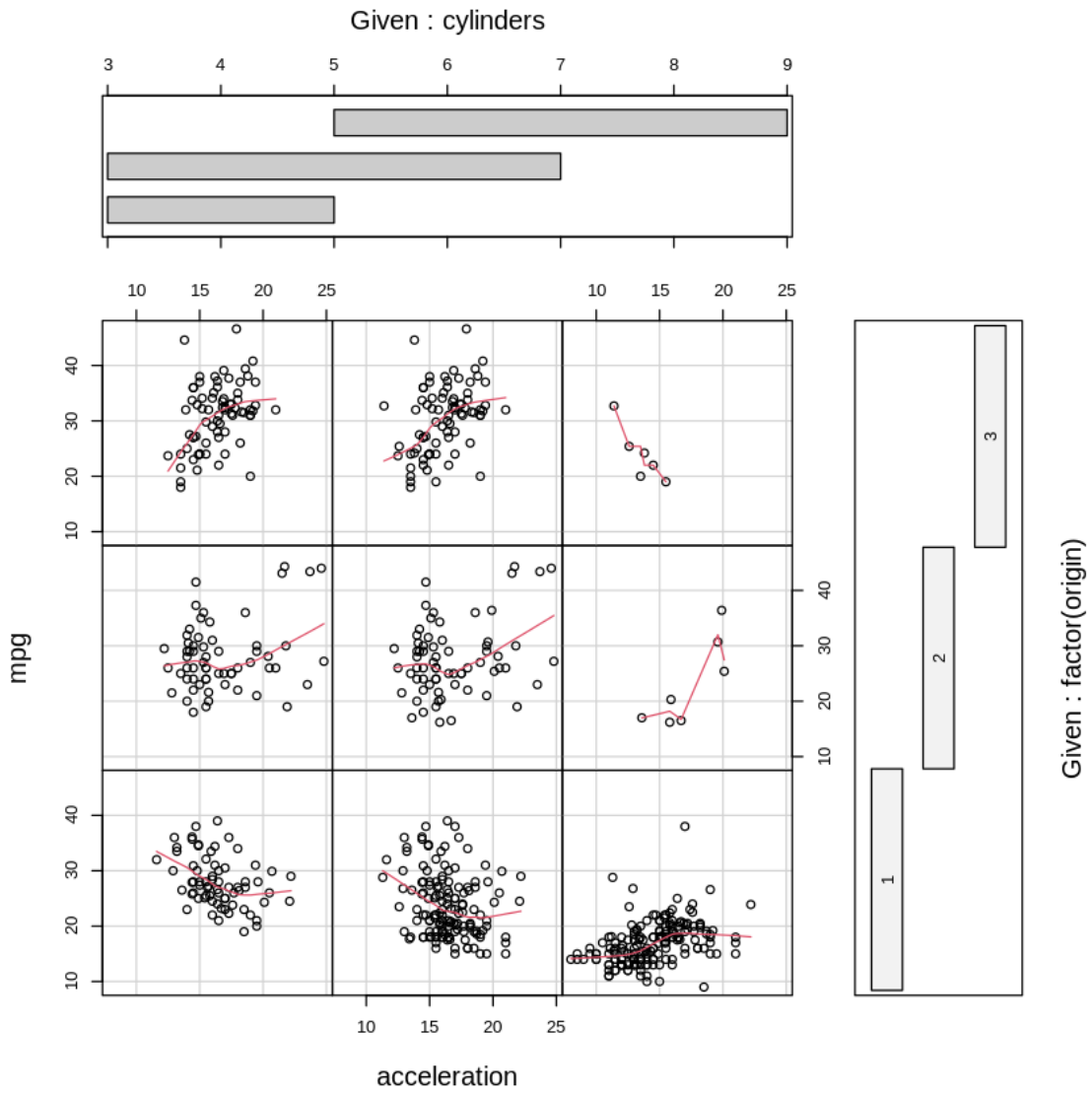
[40]:
```
coplot(mpg ~ acceleration | horsepower*factor(cylinders) ,
       number = 6, rows = 1,
        panel= panel.smooth , data = df)


# Interaction effect exists
```



[41]:
```
coplot(mpg ~ acceleration | factor(origin)*factor(cylinders) ,
        number = 6, rows = 1,
         panel= panel.smooth , data = df)


# Interaction effect exists
```

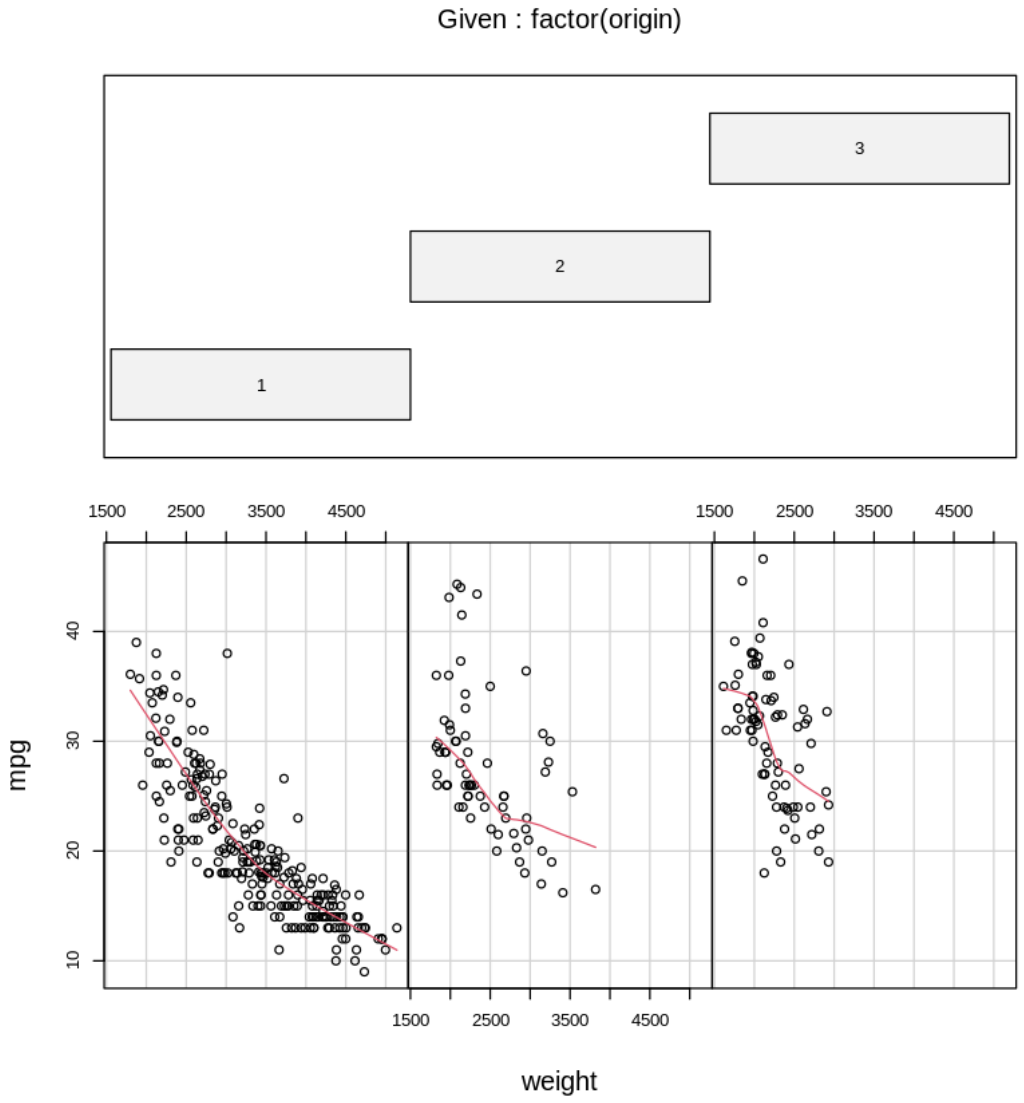


[42]:
```
coplot(mpg ~ acceleration | cylinders*factor(origin) ,
        number = 6, rows = 1,
         panel= panel.smooth , data = df)
```

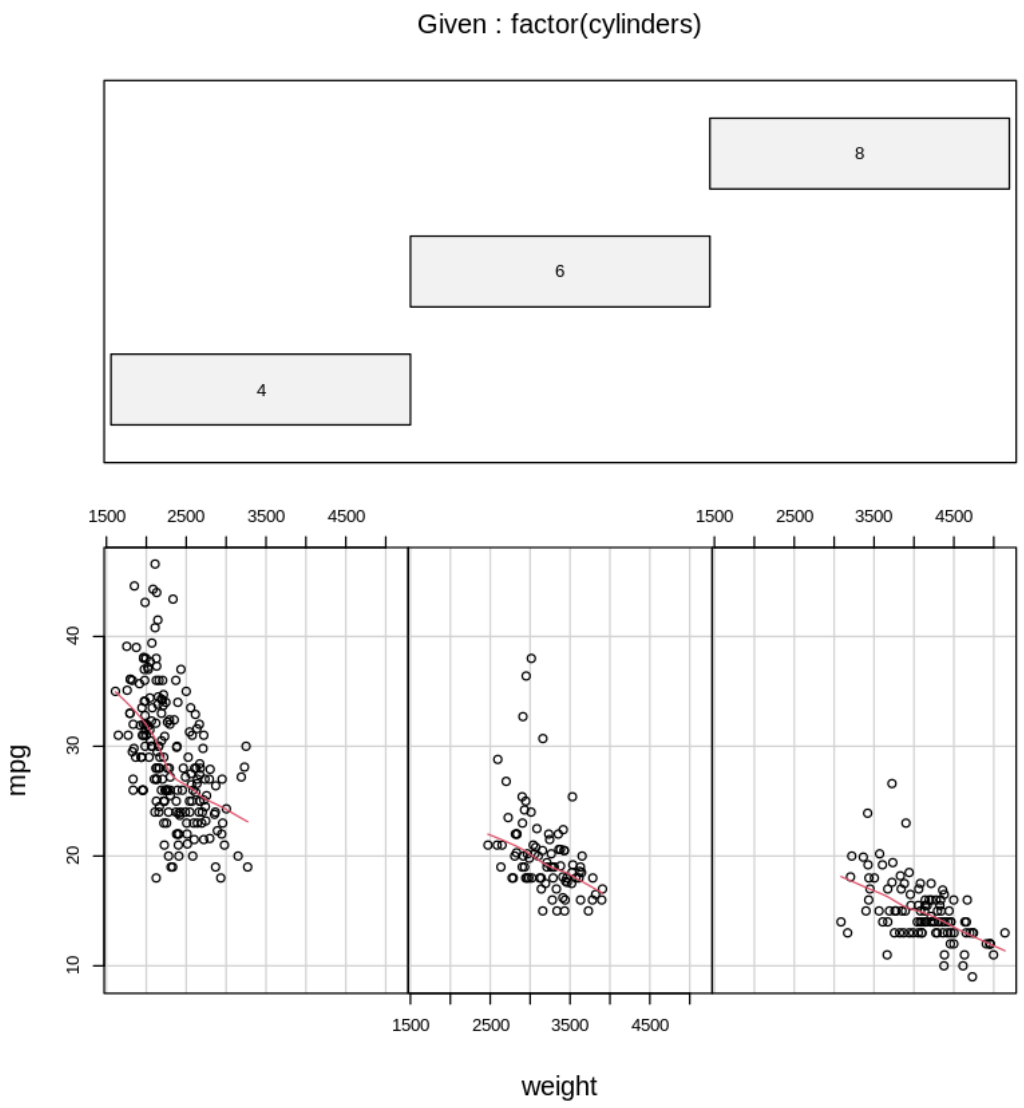


4 Weight

[43]:
```
coplot(mpg ~ weight | factor(origin) ,data=df,
       number = 4, rows = 1,
         panel = panel.smooth)

#Interaction effect exist
```



[44]:
```
coplot(mpg ~ weight | factor(cylinders) ,data=df,
       number = 4, rows = 1,
       panel = panel.smooth)

#seem to exist
```



#As hp, weight and displacement are highly correlated, interaction in weight results in others too.

5 Answer 1: On the basis of Univariate plots and Coplots

1) MPG decreases with increase in number of cylinders.

2) MPG is negatively associated with displacement and the relation is non linear as per the EDA.

3) MPG is negatively associated with horsepower and the relationship in non linear.

4) MPG has a positive correlation with acceleration and the relation appears to be linear as per the plots.

5) MPG increases with change in origin from 1(American) to 2(European) but nearly remains

same for 2(European) to 3(Japanese). The effect is attributed to the more number of cylinders in american cars.

6) From the pair plot of the variables we can see that Horsepower, Displacement and Weight are significantly correlated this is very intuitive because larger engines would have more horsepower and weight of the car would be more due to engine size.

From a mechanical perspective , things tends to fill in place. As number of cylinders increase, weight increases and also the volume displaced by the piston increases which results in horsepower increase. This leads to lower time period for acceleration of the vehicle from 0 to t velocity.

Interaction Effects(2nd Order)

1) Interaction seems to be significant for Origin 1(America) and Horsepower as similar pattern is not seen for origin 2 and origin 3.

2) Interaction effect is present for horsepower and Number of cylinders and also for log horsepower and Number of Cylinders that is very intuitive because number of cylinders will affect how the horsepower affects mpg.

3) As already stated Diplacement, Weight and Horsepower are correlated significantly, similar kind of interaction of these with other variables seem to exist.

Interaction Effects(3rd Order)

1) Acceleration and Weight/Displacement/Horsepower and Origin show a high level of interaction as per the Coplots incorporating those maybe helpful for analysis.

2) Acceleration and Weight/Displacement/Horsepower and Number of Cylinders show interaction effect that seems significant as per the coplots.

[53]:
```
# We are clubbbing japanese and european cars as in the EDA most of their␣
 ↪attributes appear to be same
df$origin[df$origin == 2] <- 0
df$origin[df$origin == 3] <- 0
```

[54]:
```
# Including all the non-linearity and Interaction term findings in df to make␣
 ↪further application of feature selection easier.

df$log.hp =log(df$horsepower)
df$log.weight =log(df$weight)
df$log.displacement = log(df$displacement)
df$acc.origin = df$acceleration*df$origin
df$acc.weight = df$acceleration*df$weight
df$acc.hp = df$acceleration*df$horsepower
df$acc.weight.origin = df$acceleration*df$weight*df$origin
df$acc.disp.origin = df$acceleration*df$displacement*df$cylinders
df$acc.hp.origin = df$acceleration*df$horsepower*df$origin
df$acc.hp.cyl = df$acceleration*df$horsepower*df$cylinders
df$hp.origin = df$horsepower*df$origin
df$hp.cyl = df$horsepower*df$cylinders
```

```
df$disp.origin = df$displacement*df$origin
df$dist.cyl = df$displacement*df$cylinders
df$weight.origin = df$weight*df$origin
df$weight.cyl = df$weight*df$cylinders

# Omitting name from the df
df$name = NULL
```

[55]:
```
# Implementing full model on my expected variables as per the EDA and␣
 ↪Interaction Plot analysis

# Define the linear regression model with all independent variables
model <- lm( log(mpg) ~ cylinders + displacement + horsepower + weight + year +␣
 ↪origin + sqrt(weight)  + sqrt(horsepower) + sqrt(displacement) +
              log.hp + log.displacement + log.weight + acc.weight.origin + acc.
 ↪disp.origin + acc.origin + acc.weight + acc.hp +
              acc.hp.origin + acc.hp.cyl + hp.origin + hp.cyl +
              disp.origin + dist.cyl + weight.origin + weight.cyl, data = df)
```

[56]:
```
# Implementing recursive feature selection based on p - value
bothfit.val<-ols_step_both_aic(model)
bothfit.val
```

```
                                Stepwise Summary
-------------------------------------------------------------------------------------
Variable              Method       AIC        RSS       Sum Sq     R-Sq       Adj.␣
 ↪R-Sq
-------------------------------------------------------------------------------------
sqrt(weight)          addition    -303.033    10.434    34.776    0.76922      0.
 ↪76863
year                  addition    -551.984     5.501    39.709    0.87833      0.
 ↪87771
log.hp                addition    -568.337     5.249    39.961    0.88390      0.
 ↪88300
acc.weight.origin     addition    -591.375     4.924    40.286    0.89108      0.
 ↪88996
weight.origin         addition    -607.534     4.701    40.509    0.89601      0.
 ↪89466
cylinders             addition    -611.470     4.631    40.579    0.89757      0.
 ↪89598
acc.disp.origin       addition    -613.828     4.579    40.631    0.89871      0.
 ↪89686
disp.origin           addition    -615.883     4.532    40.678    0.89975      0.
 ↪89766
```

```
acc.hp.cyl            addition     -617.536     4.490     40.720     0.90068      0.
↪89834
hp.origin             addition     -624.194     4.392     40.818     0.90285      0.
↪90030
acc.hp.origin         addition     -625.572     4.355     40.855     0.90368      0.
↪90089
acc.weight.origin     removal      -627.044     4.360     40.850     0.90355      0.
↪90102
--------------------------------------------------------------------------------
```

[57]:
```
# Implementing recursive feature selection based on p - value
bothfit.p<-ols_step_both_p(model, pent = 0.05,p_remove=.1)
bothfit.p
```

```
                                Stepwise Selection Summary
-------------------------------------------------------------------------------------------
                                 Added/                   Adj.
Step        Variable            Removed     R-Square     R-Square      C(p)          ␣
↪AIC         RMSE
-------------------------------------------------------------------------------------------
   1      sqrt(weight)          addition       0.769        0.769     532.8380     ␣
↪-303.0328    0.1636
   2          year              addition       0.878        0.878      99.4620     ␣
↪-551.9844    0.1189
   3         log.hp             addition       0.884        0.883      79.2580     ␣
↪-568.3370    0.1163
   4    acc.weight.origin       addition       0.891        0.890      52.5940     ␣
↪-591.3747    0.1128
   5      weight.origin         addition       0.896        0.895      34.9200     ␣
↪-607.5343    0.1104
   6        cylinders           addition       0.898        0.896      30.6850     ␣
↪-611.4702    0.1097
   7     acc.disp.origin        addition       0.899        0.897      28.1660     ␣
↪-613.8281    0.1092
   8       disp.origin          addition       0.900        0.898      26.0070     ␣
↪-615.8829    0.1088
-------------------------------------------------------------------------------------------
```

We tried models from Recursive selection but their diagonistics were failing so tried different model by trial and error that eventually passed all the diagonistics. We have included log terms and root terms and other interaction terms also in the full model that appeared to be signinficant from the plots.

[58]:
```
model2 <- lm( log(mpg) ~ log(weight)  + year  + log.hp  + I(origin  * log(␣
↪acceleration) ), data = df)
```

```
summary(model2)
```

```
Call:
lm(formula = log(mpg) ~ log(weight) + year + log.hp + I(origin *
    log(acceleration)), data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-0.37435 -0.07163  0.00290  0.06813  0.38825

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                    7.189455   0.284835  25.241  < 2e-16 ***
log(weight)                   -0.704180   0.046679 -15.085  < 2e-16 ***
year                           0.030774   0.001715  17.949  < 2e-16 ***
log.hp                        -0.172557   0.036451  -4.734 3.09e-06 ***
I(origin * log(acceleration)) -0.019605   0.005441  -3.603 0.000356 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1134 on 387 degrees of freedom
Multiple R-squared:  0.8898,        Adjusted R-squared:  0.8887
F-statistic: 781.4 on 4 and 387 DF,  p-value: < 2.2e-16
```

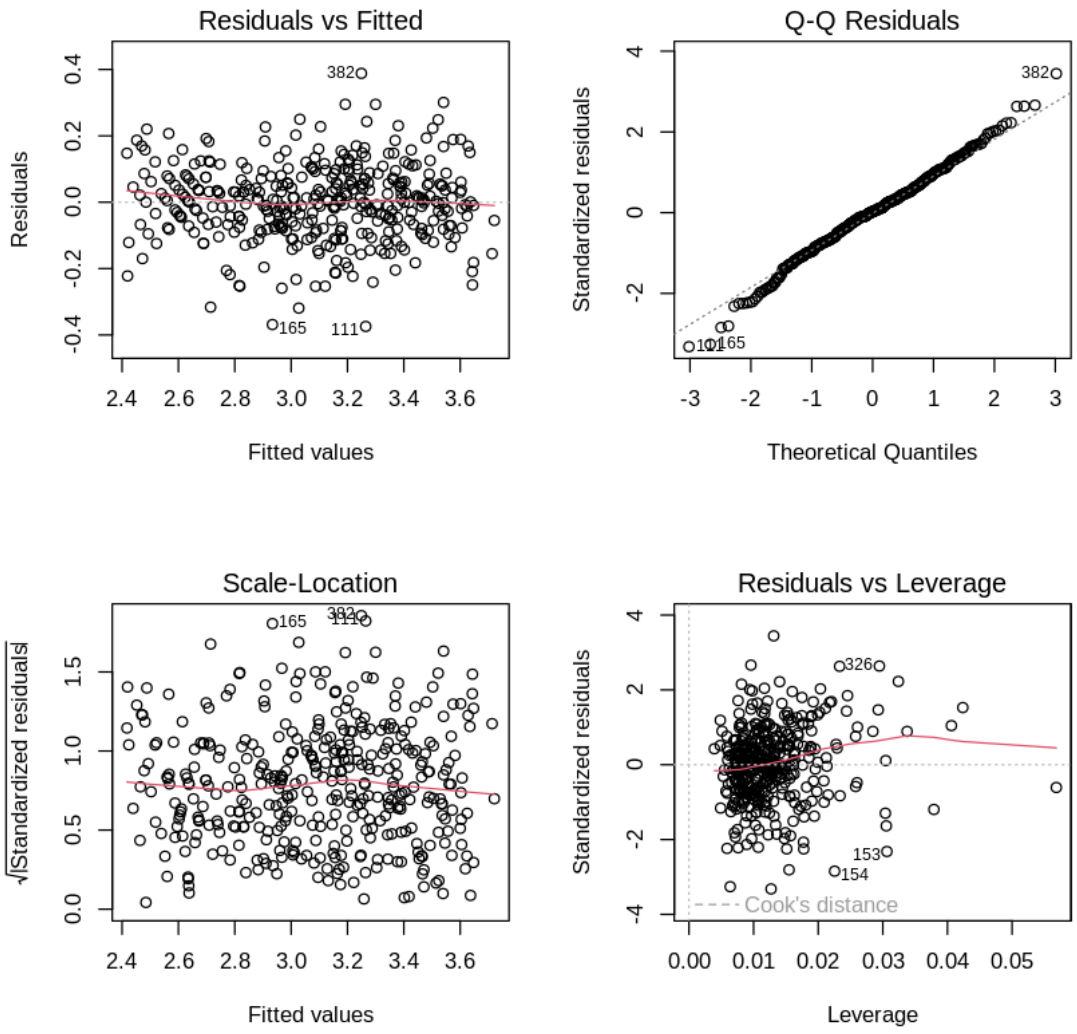
6 Question 2: Here model 2 is our chosen model because it passed all the diagonistic tests of the linear regression. But the thing worth noticing is that none of our Recursive model were satisfying assumptions of linear regression. We came to this model 2 by trial with different combination of variables.

Our model regresses log(mpg) as a function of log(weight), year, log hp and interaction of origin and log acceleration. Some of the important variables from engineering point of view seem to have vanished but for the sake of validation of assumption we bear it.

[59]:
```
rse = sqrt(deviance(model2)/df.residual(model2))
sres2 <-residuals(model2)/(rse*sqrt(1-influence(model2)$hat))
shapiro.test(sres2)

# Perform Kolmogorov-Smirnov test for normality
ks.test(sres2, "pnorm", mean(sres2), sd(sres2))

#Test for homoskedascticity i.e. Breusch-Pagan test
bptest(model2)
```

```r
# Set up a 2x2 grid for diagnostic plots
par(mfrow = c(2, 2))

# Plot diagnostic plots for the model
plot(model2)
```

```
        Shapiro-Wilk normality test

data:  sres2
W = 0.99356, p-value = 0.0939


        Asymptotic one-sample Kolmogorov-Smirnov test

data:  sres2
D = 0.039495, p-value = 0.5738
alternative hypothesis: two-sided


        studentized Breusch-Pagan test

data:  model2
BP = 5.524, df = 4, p-value = 0.2376
```



7 Question 3: Model Diagonistics

Here we have tested normality using Shapiro Wilk and KS test both established normality, Breush Pagan test for Homoskedasticity that was validated and also the residual plots seems to be random no visible pattern is present, there are no outliers as per the plots. Thus from our residual analysis and statistical tests we can say that our model is tenable as all the assumptions are validated and we are good to go with this model.

8 Question 4: Based on model the association of variables

1. MPG is negatively associated with the weight of the car keeping other terms constant. This is very intuitive also because heavier vehicles tend to consume more fuel so lower mileage.

2. MPG is positive associated with the year of manufacturing of the car that can be attributed to the fact that we have new innovations in the cars as the time passes which leads to better mileage.
3. MPG is negatively associated with horsepower keeping other things constant. The reason for this is attributed to the marginal inverse relation between MPG and Horsepower. From engineering point of view because vehicles of more horsepower consume more fuel per km generally.
4. The interaction term between log acceleration and origin has a negative association with the MPG. It boils down to the american cars having less MPG for a given acceleration as compared to european and japanese cars. This decrease in MPG with increase in acceleration is clear from the EDA for the American cars but the same effect does not seem profound for Japanese and European cars.

At last, we selected a model that is satisfying all the assumptions but with some more transformation terms. We fealt that validating the assumptions of the model is more important along with maintaining its good performance that is R squared of 0.8898 that is comparable to other step selection approaches.

[52]: