Functions to Automate Preliminary Bivariate Analyses

Mike Kirkpatrick

December 22, 2017

Overview

This code is designed to expedite the preliminary bivariate comparisons that are usually done prior to statistical modeling.

For example, if you are going to run a multiple regression model, you likely will test each independent variable against your dependent variable to see if there is a statisticaly significant relationship. Additionally, you will probably create a plot visualizing the statistically significant relationships.

These exploratory analyses, although important, can become painfully monotonous. Therefore, I wrote this code to automate this process for **continuous** and **dichotomous** dependent variables. This file illustrates how the code could be used on the `mtcars` data set.

Data Preparation

First I simply import and then view the mtcars (https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/mtcars.html) dataset.

```
data(mtcars)
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Then I create an analysis data set (`df`). I change `cyc` to a categorical factor variable. I recode the variables `am` and `vs` and make them categorical factor variables.

```
df <- mtcars

df$cyl <- as.factor(df$cyl)

df$am <- as.factor(
  sapply(df$am, function(i)
  if (i == 0) {"automatic"}
  else if (i==1) {"manual"}
))

df$vs <- as.factor(
  sapply(df$vs, function(i)
    if (i == 0) {"V-engine"}
    else if (i==1) {"Straigt engine"}
  ))

head(df)
```

```
##                    mpg cyl disp  hp drat    wt  qsec             vs
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46       V-engine
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02       V-engine
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61 Straigt engine
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44 Straigt engine
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02       V-engine
## Valiant           18.1   6  225 105 2.76 3.460 20.22 Straigt engine
##                          am gear carb
## Mazda RX4            manual    4    4
## Mazda RX4 Wag        manual    4    4
## Datsun 710           manual    4    1
## Hornet 4 Drive    automatic    3    1
## Hornet Sportabout automatic    3    2
## Valiant           automatic    3    1
```

Set Default Parameters

I used only base R packages except for the `car` package, which is needed to get the ANOVA p-values for dichotomous DVs. I chose to set my `alpha` level to 0.05 for determining statistical significance. The value I choose for `alpha` here doesn't affect the actual statistical tests. Rather, it simply changes the label the IV is given in the summary table (more on that later). The `yrange` function makes the y-axis of charts 25% bigger and `color` simply defines two colors to be used in boxplots and barplots.

```
library(car)
alpha <- 0.05
yrange <- function(y) c(0,max(y)*1.25)
color <- c("coral","aquamarine3")
```

Continuous Dependent Variable

The following sections are to be used if your dependent variable is **continuous**.

Table of Statistical Tests

I choose `mpg` as my dependent variable (`DV`). I then create a list of all other variables in my data set and considers them as the independent variables (`IVs`).

```
DV <- "mpg"
IVs <- colnames(df)[colnames(df)!= DV]
```

I then define a function that will do the following to each DV and IV pair:

1. Create a **linear** regression model such that DV ~ IV
2. Create an ANOVA table
3. Create a summary table
4. Extract the degrees of freedom, F-statistic, p-value, r-squared, and put all of these elements into a data frame (table) called "r"
5. Add the name of the DV and IV to the table
6. Determine if the IV is statistically significant based on the `alpha` level I chose previsouly. Add this label to the table
7. Reset the `rownames`

```
f.ConDV <- function(IV) {
  m <- lm (df[,DV] ~ df[,IV])
  a <- anova(m)
  s <- summary(m)
  r <- data.frame(cbind(df1=s$fstatistic[2], df2=s$fstatistic[3], F.stat=round(s$fstatistic[1],2
), p.value=round(a$`Pr(>F)`[1],3), r.sqr = round(s$r.squared,2)))
  r <- cbind(DV,IV,r)
  r$Sig <- ifelse(r$p.value < alpha,'Yes','No')
  rownames(r) <- NULL
  return(r)
}
```

Now, I apply my function to my data (line 1). I order my table of statistical tests by the p-value and F-statistic. I reset the rownames to preserve this order and then I display the results.

NOTE: This table contains the results of 10 *separate* linear regression models. This table does NOT contain an ANOVA table for a linear regression model that includes all 10 of these IVs.

The results indicate that all of the IVs are statistically significant predictors of `MPG` . The IV `wt` has the strongest relationship with `mpg` , and `qsec` has the weakest.

```
tbl.ConDV <- do.call(rbind,lapply(IVs, f.ConDV))
tbl.ConDV <- tbl.ConDV[order(tbl.ConDV$p.value,-tbl.ConDV$F.stat),]
rownames(tbl.ConDV) <- NULL
print(tbl.ConDV)
```

```
##     DV   IV df1 df2 F.stat p.value r.sqr Sig
## 1  mpg   wt   1  30  91.38   0.000  0.75 Yes
## 2  mpg disp   1  30  76.51   0.000  0.72 Yes
## 3  mpg   hp   1  30  45.46   0.000  0.60 Yes
## 4  mpg  cyl   2  29  39.70   0.000  0.73 Yes
## 5  mpg drat   1  30  25.97   0.000  0.46 Yes
## 6  mpg   vs   1  30  23.66   0.000  0.44 Yes
## 7  mpg   am   1  30  16.86   0.000  0.36 Yes
## 8  mpg carb   1  30  13.07   0.001  0.30 Yes
## 9  mpg gear   1  30   9.00   0.005  0.23 Yes
## 10 mpg qsec   1  30   6.38   0.017  0.18 Yes
```

Plots of Statistically Significant IVs

Now that I know which IVs are significant predictors of my DV, I want to visualize each bivariate relationship. To do this, I make sure my DV is still defined. Then I get a list of my statistically significant IVs.

```
DV <- "mpg"
IVs <- as.matrix(subset(tbl.ConDV,Sig=="Yes",select = IV))
```

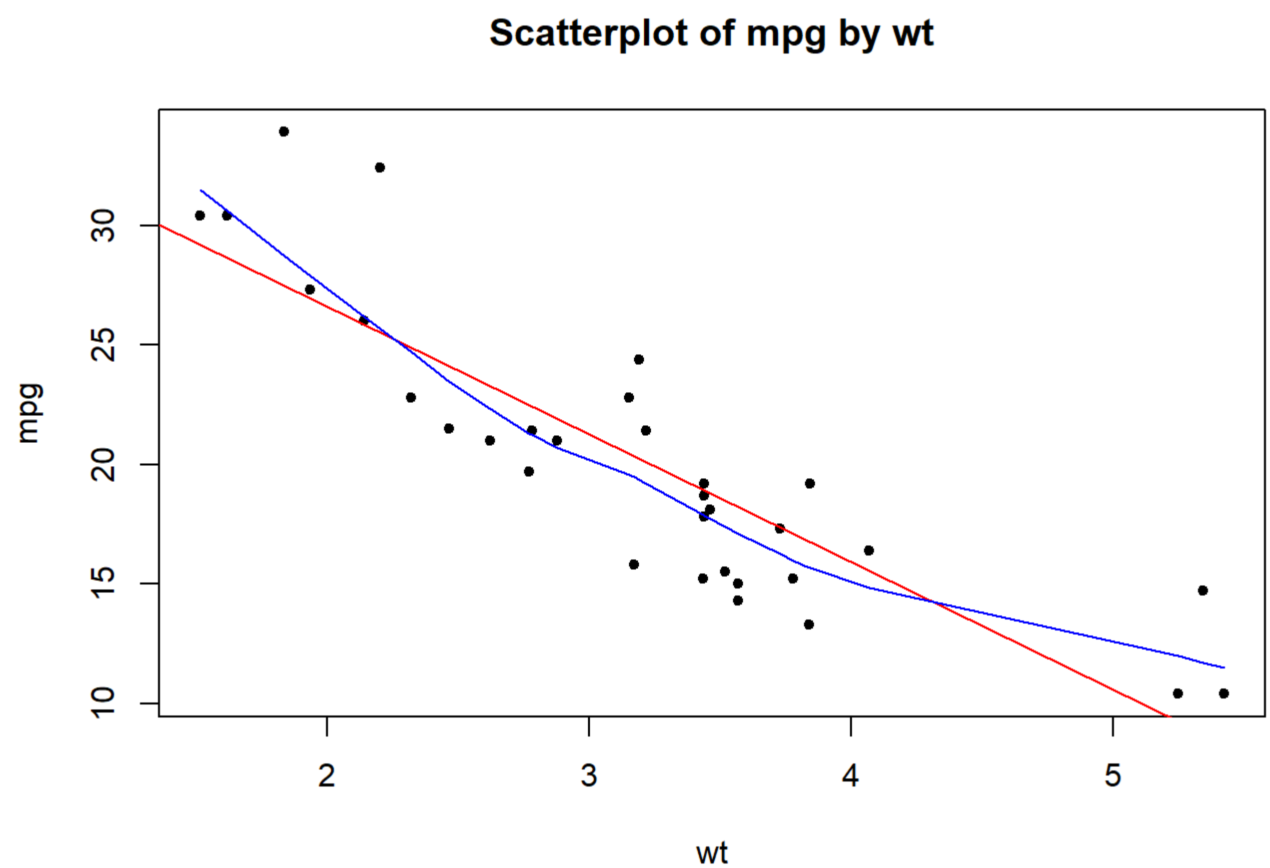
I then define a function that will do the following to each DV and IV pair:

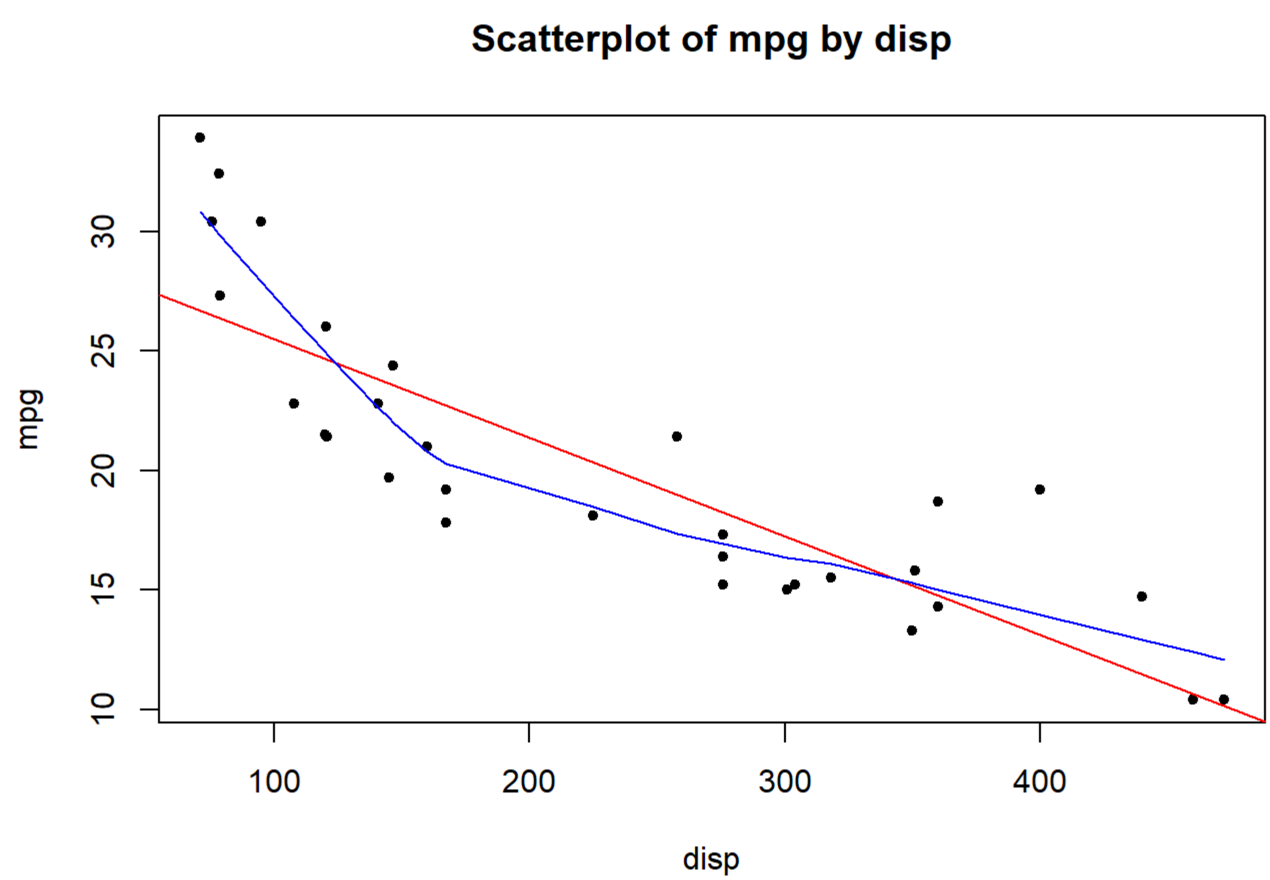
1. Determine if the IV is continuous (“numeric”) or categorical (“factor”)
2. If the IV is continuous, create a scatterplot
3. If the IV is categorical, create a boxplot

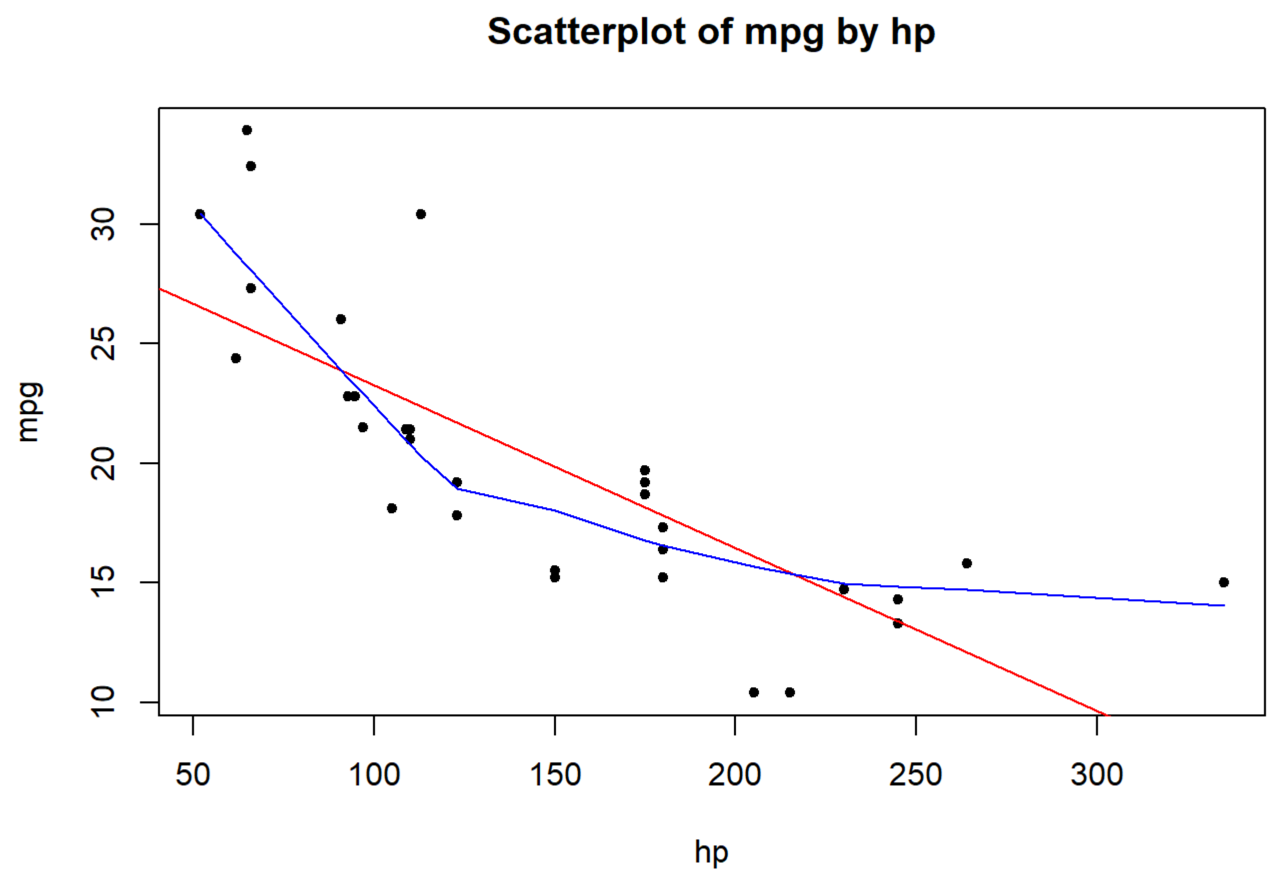
```
f.ConDvPlots <- function(IV) {
  if(class(df[,IV])=="numeric") {
    plot(df[,IV],df[,DV], main=paste("Scatterplot of",DV,"by",IV), xlab=IV, ylab=DV, pch=20)
    abline(lm(df[,DV]~df[,IV]), col="red")
    lines(lowess(df[,IV],df[,DV]), col="blue")
  }
  else if (class(df[,IV])=="factor") {
    boxplot(df[,DV] ~ df[,IV], col = color, main=paste("Boxplot of",DV,"by",IV), ylab=DV, xlab=I
V)
    }
}
```

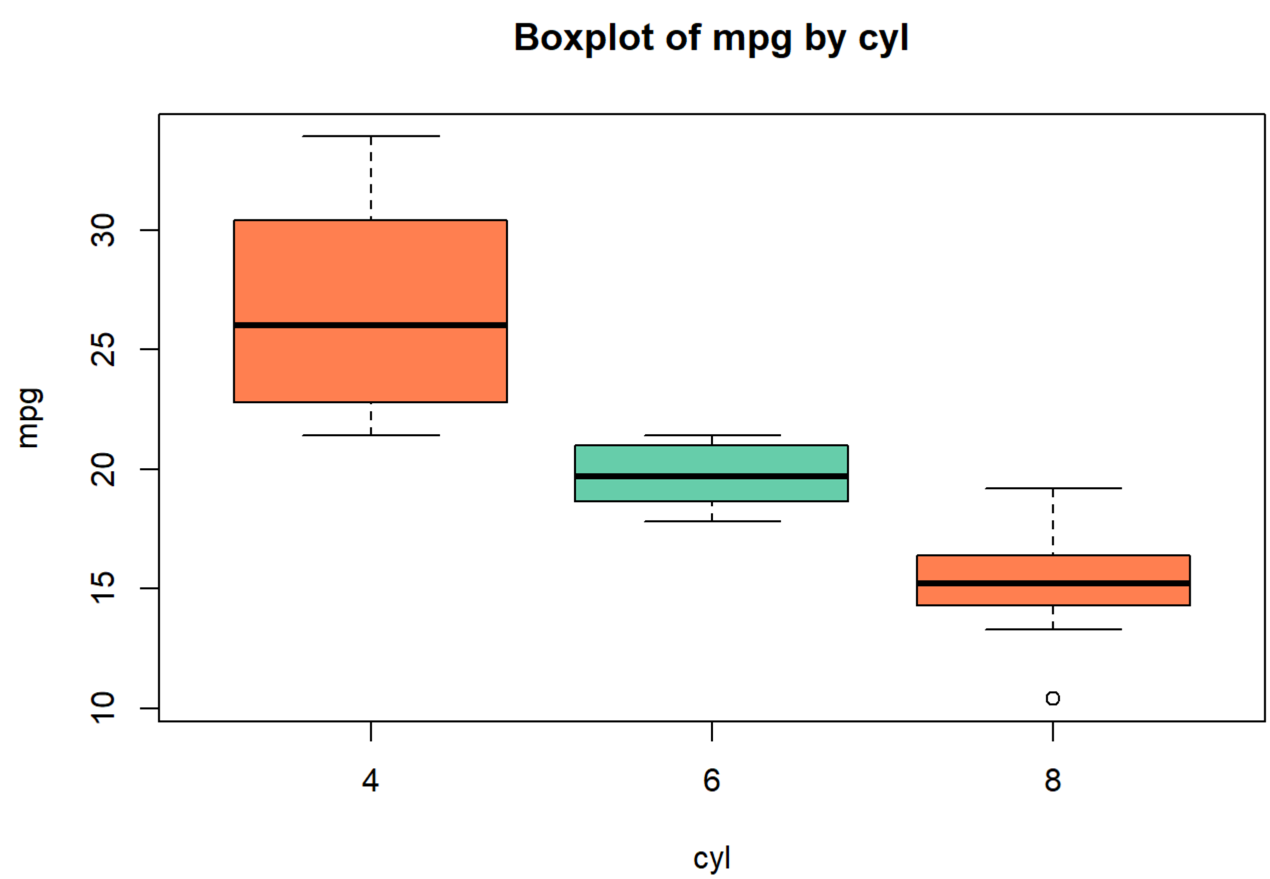
I execute the function by simply running the following.

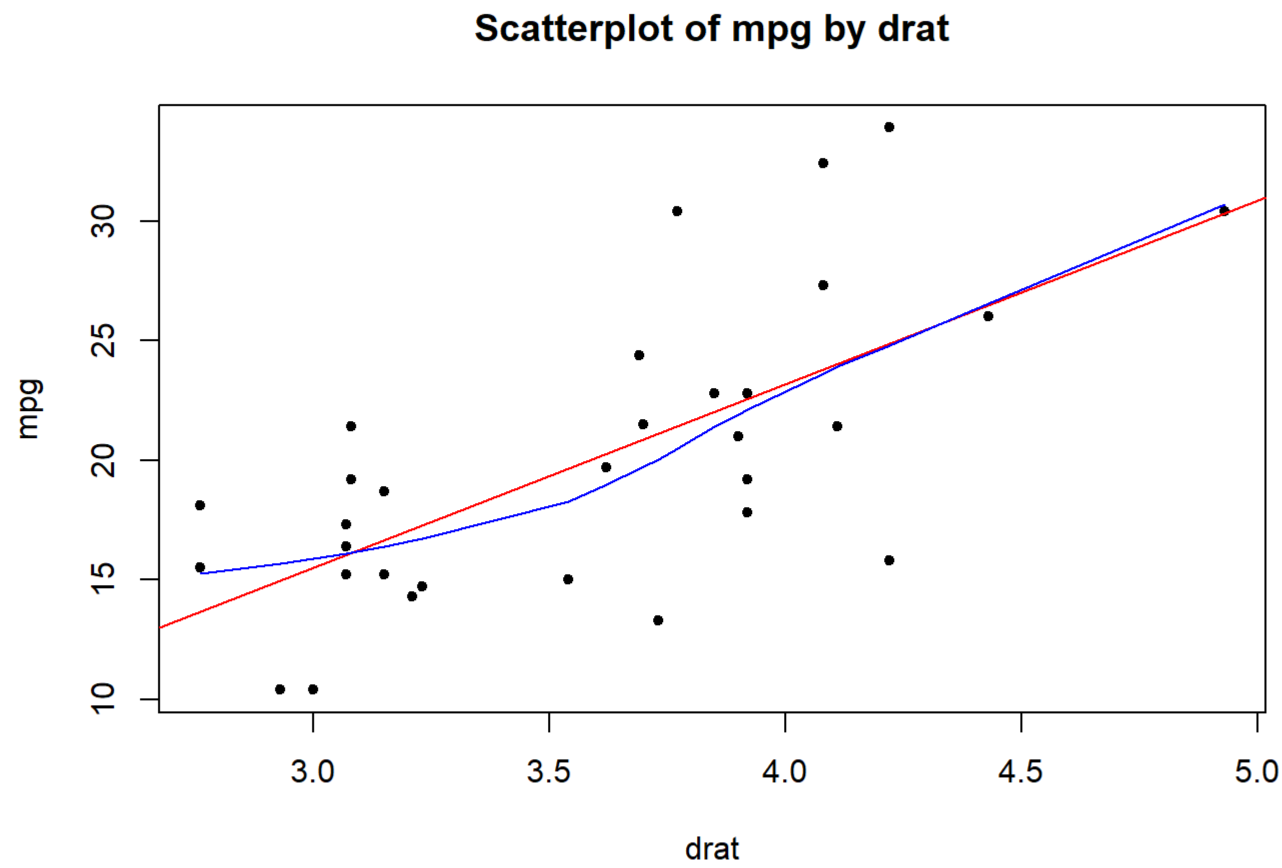
```
lapply(IVs, f.ConDvPlots)
```

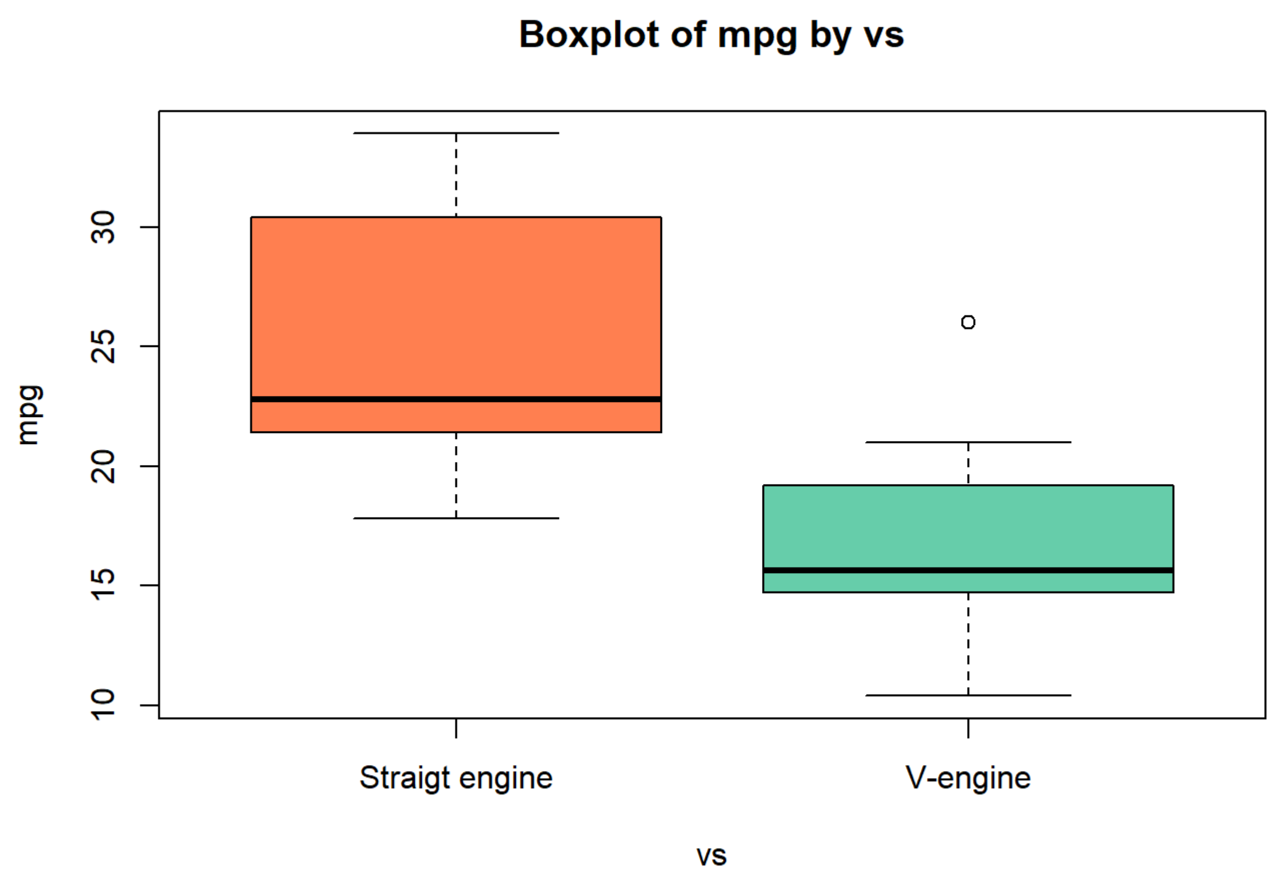

Scatterplot of mpg by wt
mpg
10
15
20
25
30
2
3
4
5
wt

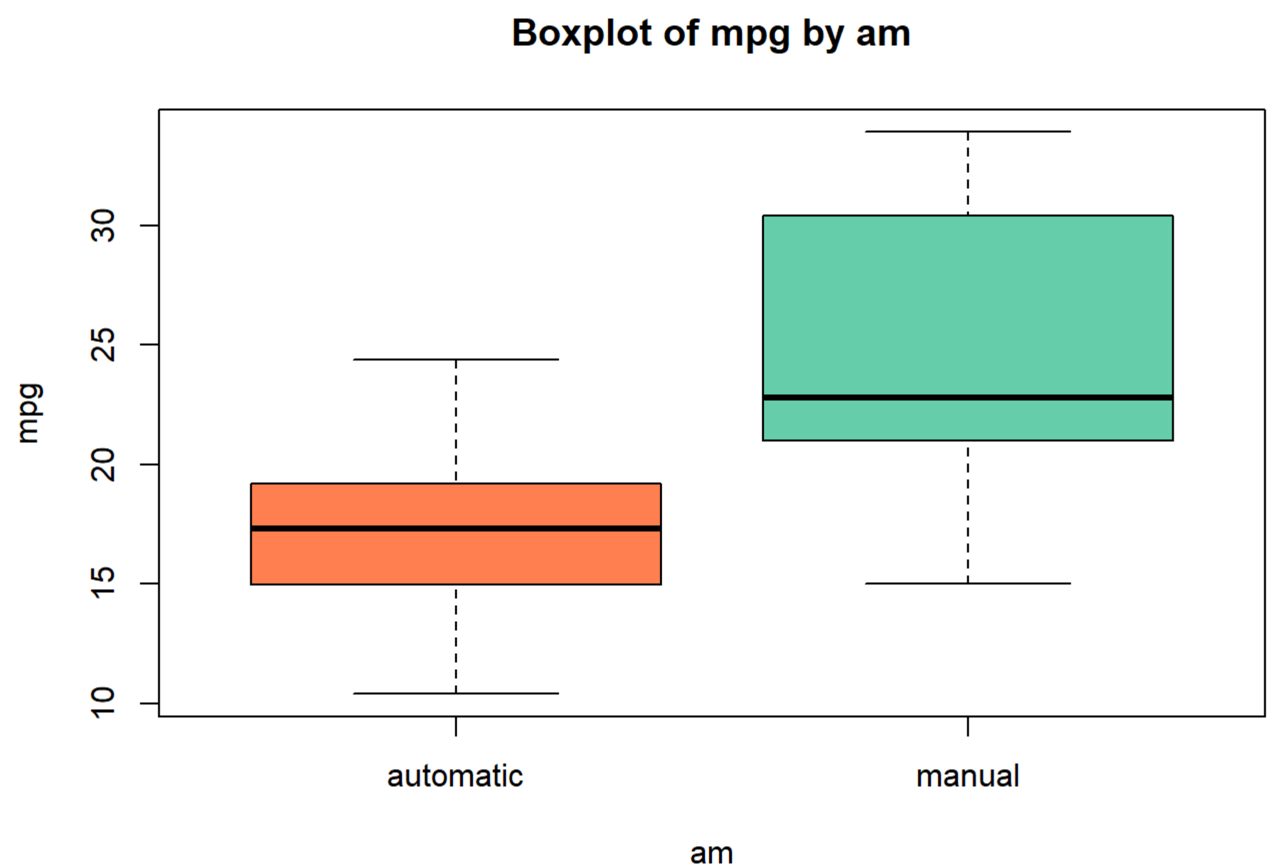

Scatterplot of mpg by disp
mpg
10
15
20
25
30
100
200
300
400
disp

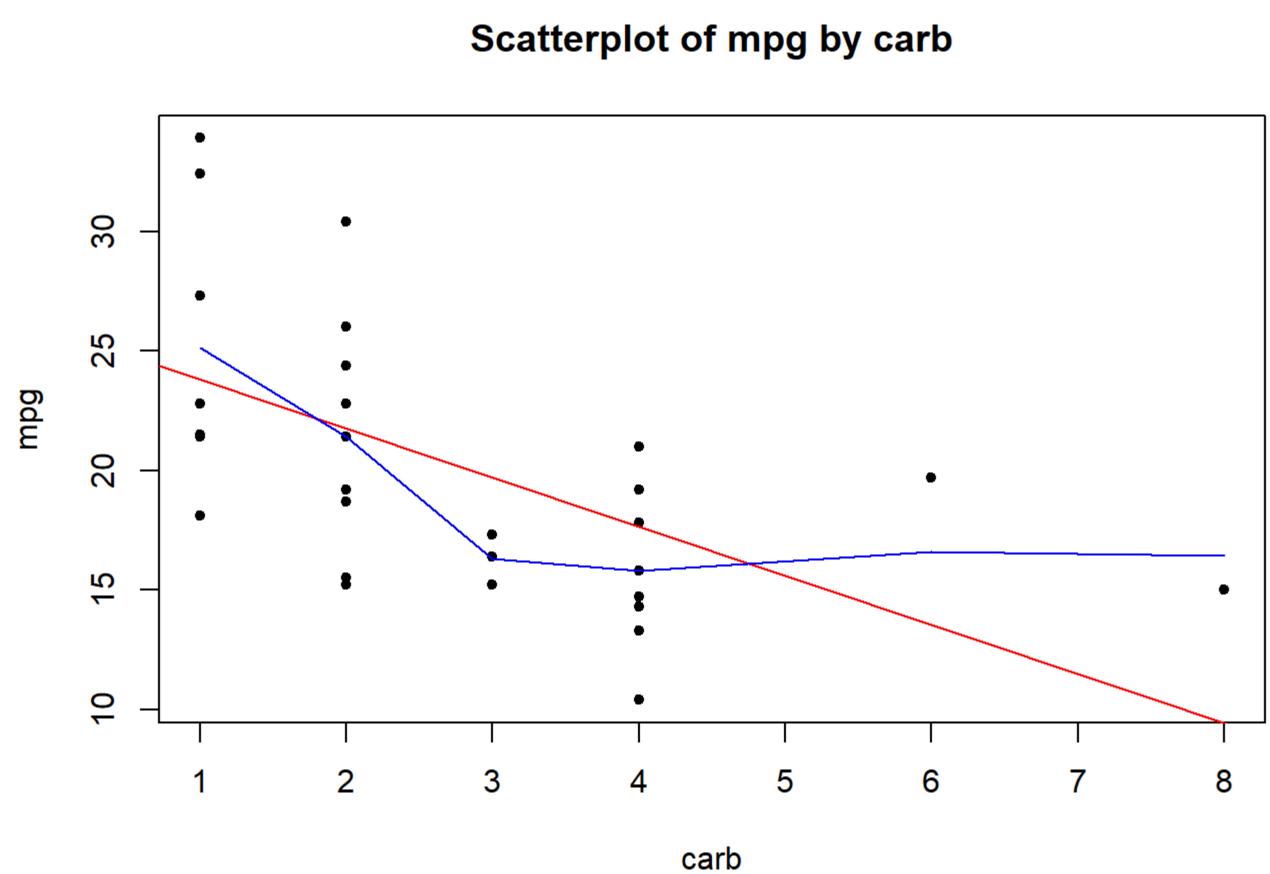

Scatterplot of mpg by hp
mpg
10
15
20
25
30
50
100
150
200
250
300
hp


Boxplot of mpg by cyl
mpg
10
15
20
25
30
4
6
8
cyl

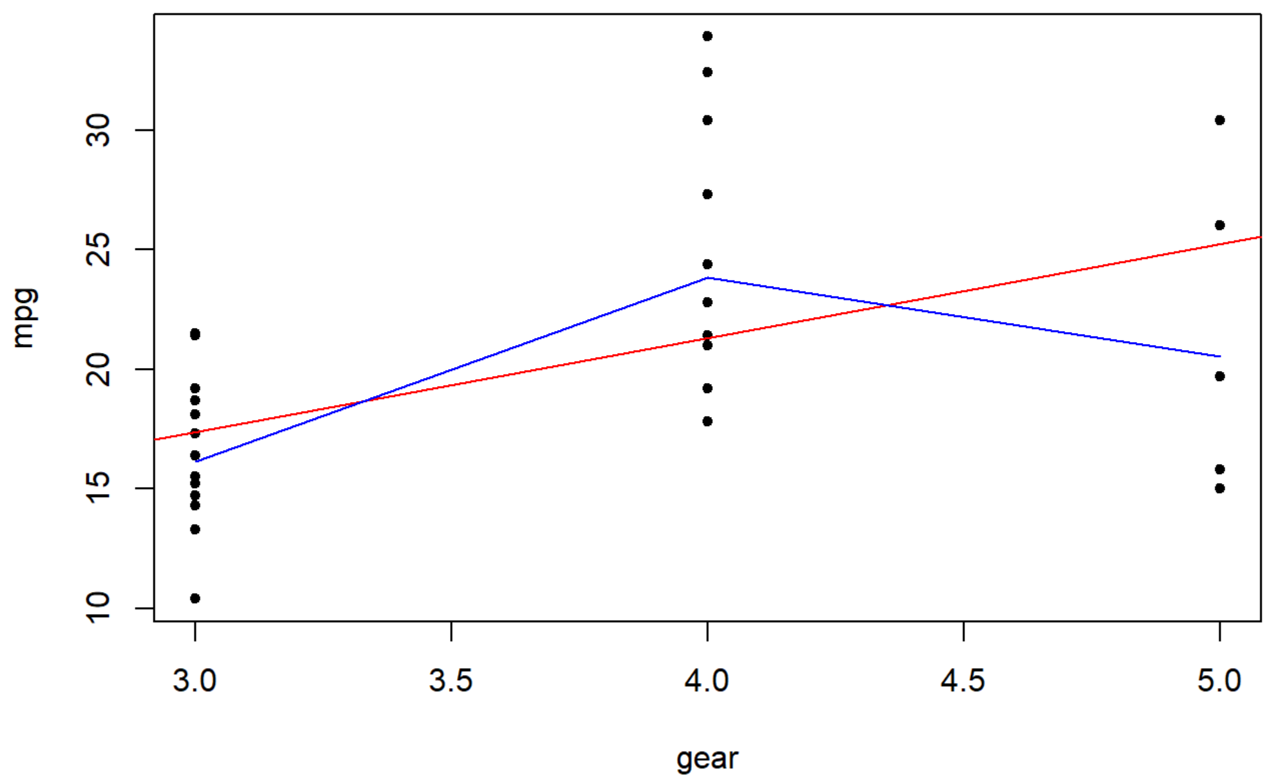

Scatterplot of mpg by drat
mpg
10
15
20
25
30
3.0
3.5
4.0
4.5
5.0
drat


Boxplot of mpg by vs
mpg
10
15
20
25
30
Straigt engine
V-engine
vs

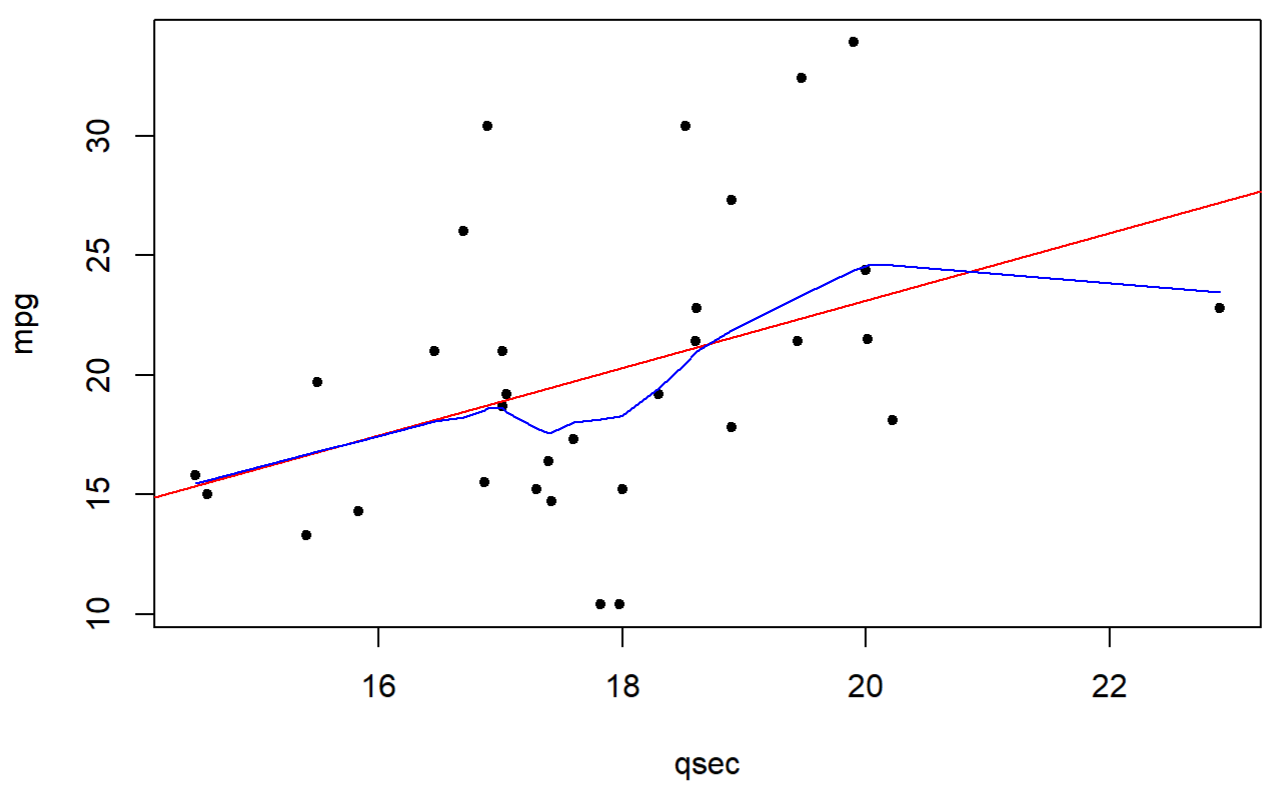

Boxplot of mpg by am
mpg
10
15
20
25
30
automatic
manual
am


Scatterplot of mpg by carb
mpg
10
15
20
25
30
1
2
3
4
5
6
7
8
carb

Scatterplot of mpg by gear



Scatterplot of mpg by qsec



Dichotomous Categorical Dependent Variable

The following sections are to be used if your dependent variable is **dichotomous**.

Table of Statistical Tests

I choose `vs` as my dependent variable (`DV`). I then create a list of all other variables in my data set and considers them as the independent variables (`IVs`).

```
DV <- "vs"
IVs <- colnames(df)[colnames(df)!= DV]
```

I then define a function that will do the following to each DV and IV pair:

1. Create a **logistic** regression model such that DV ~ IV
2. Create an ANOVA table
3. Extract the degrees of freedom, ChiSquare statistic, p-value, and put all of these elements into a data frame (table) called “r”
4. Add the name of the DV and IV to the table
5. Determine if the IV is statistically significant based on the `alpha` level I chose previsouly. Add this label to the table
6. Reset the `rownames`

```
f.CatDV <- function(IV) {
  m <- glm(df[,DV] ~ df[,IV], family = binomial)
  a <- Anova(m, type="III")
  r <- data.frame(cbind(df=a$Df, ChiSq=round(a$`LR Chisq`,2), p.value=round(a$`Pr(>Chisq)`,3)))
  r <- cbind(DV,IV,r)
  r$Sig <- ifelse(r$p.value < alpha,'Yes','No')
  rownames(r) <- NULL
  return(r)
}
```

Now, I apply my function to my data (line 1). I order my table of statistical tests by the p-value and ChiSquare statistic. I reset the rownames to preserve this order and then I display the results.

NOTE: This table contains the results of 10 *separate* logistic regression models. This table does NOT contain an ANOVA table for a logistic regression model that includes all 10 of these IVs.

The results indicate that all of the IVs except for `gear` and `am` are statistically significant predictors of `vs` . The IV `qsec` has the strongest relationship with `vs` , and `am` has the weakest.

```
tbl.CatDV <- do.call(rbind,lapply(IVs, f.CatDV))
tbl.CatDV <- tbl.CatDV[order(tbl.CatDV$p.value,-tbl.CatDV$ChiSq),]
rownames(tbl.CatDV) <- NULL
print(tbl.CatDV)
```

```
##    DV   IV df ChiSq p.value Sig
## 1  vs qsec  1 29.78   0.000 Yes
## 2  vs  cyl  2 27.60   0.000 Yes
## 3  vs   hp  1 27.02   0.000 Yes
## 4  vs disp  1 21.16   0.000 Yes
## 5  vs  mpg  1 18.33   0.000 Yes
## 6  vs carb  1 14.30   0.000 Yes
## 7  vs   wt  1 12.49   0.000 Yes
## 8  vs drat  1  6.70   0.010 Yes
## 9  vs gear  1  1.37   0.243  No
## 10 vs   am  1  0.91   0.341  No
```

Plots of Statistically Significant IVs

Now that I know which IVs are significant predictors of my DV, I want to visualize each bivariate relationship. To do this, I make sure my DV is still defined. Then I get a list of my statistically significant IVs.

```
DV <- "vs"
IVs <- as.matrix(subset(tbl.CatDV,Sig=="Yes",select = IV))
```

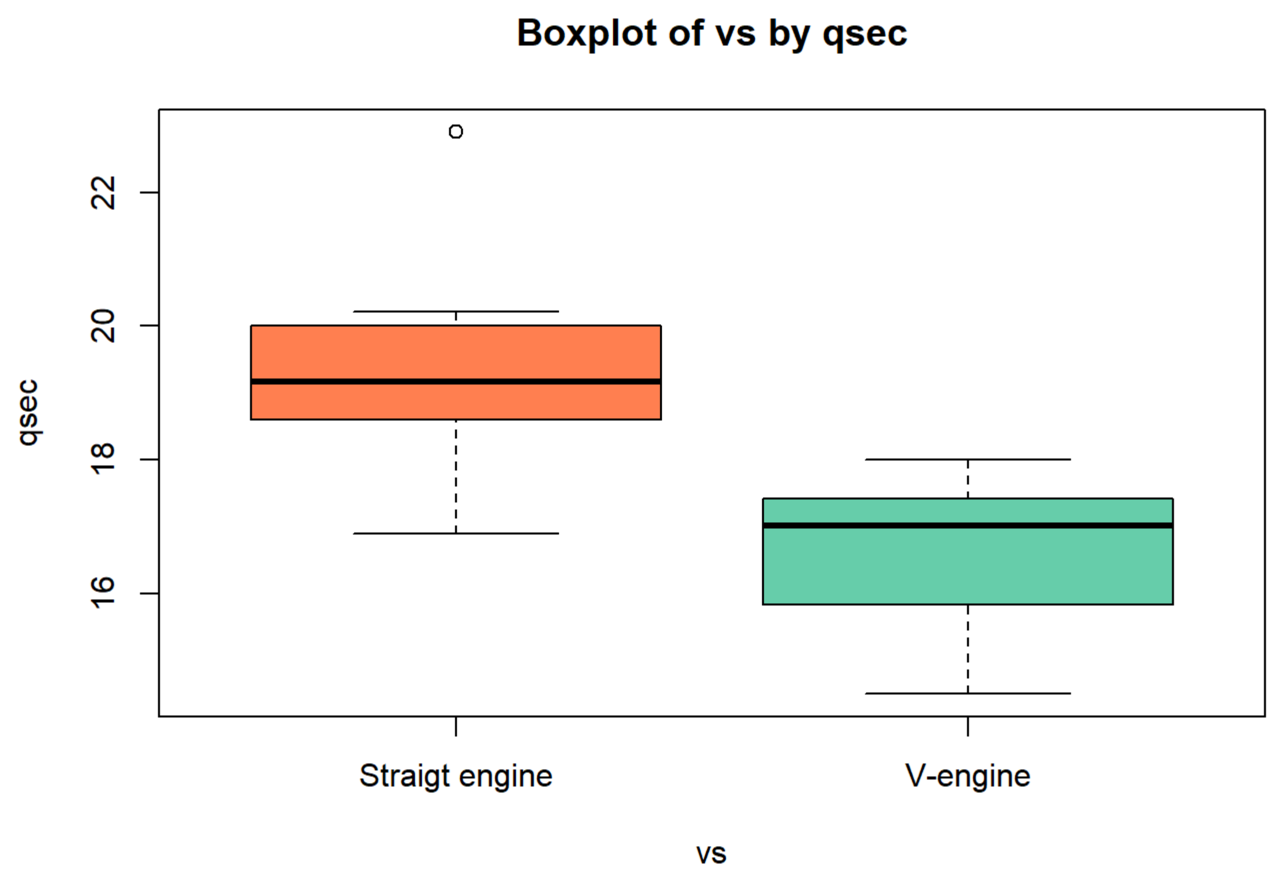
I then define a function that will do the following to each DV and IV pair:

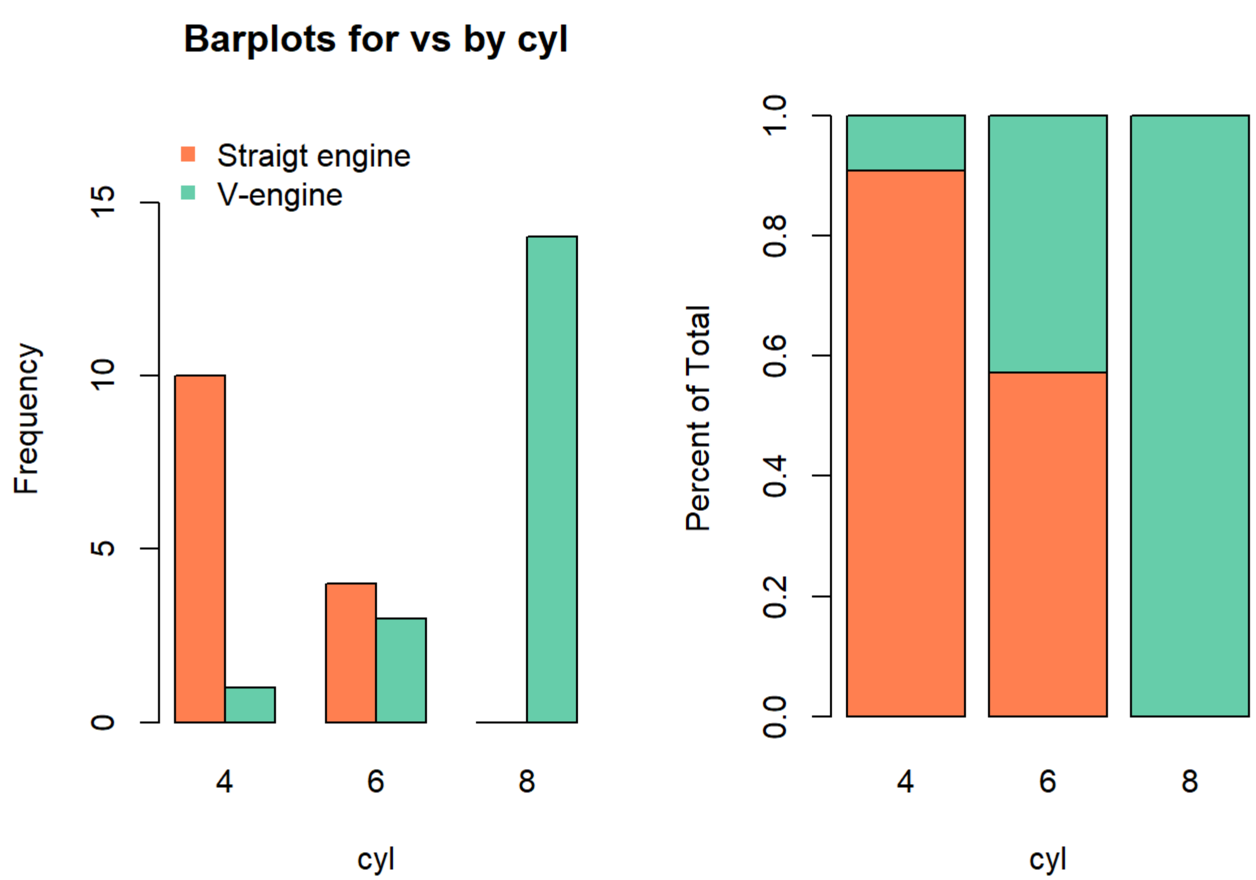
1. Determine if the IV is continuous (“numeric”) or categorical (“factor”)
2. If the IV is continuous, create a boxplot
3. If the IV is categorical, create a barplot and a stacked barplot

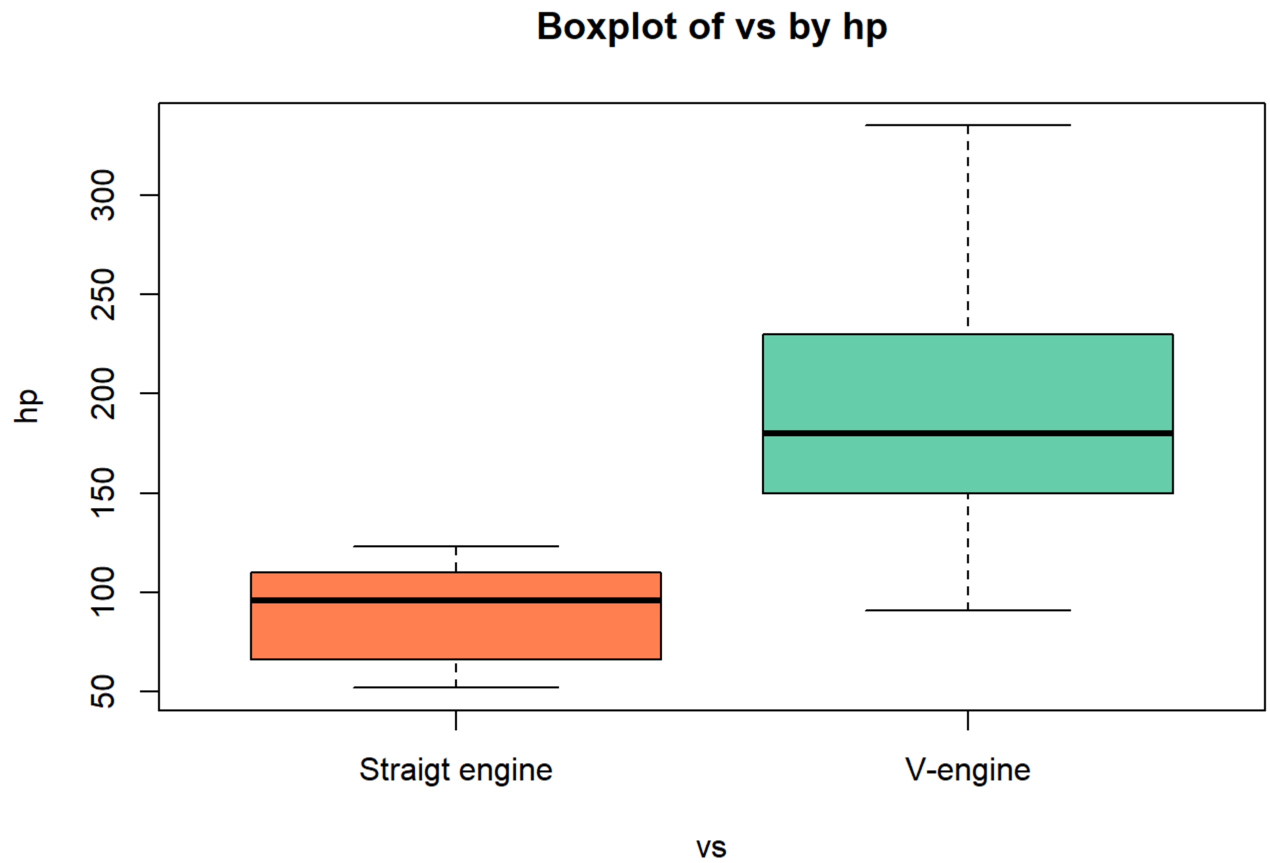
```
f.CatDvPlots <- function(IV) {
  if(class(df[,IV])=="numeric") {
    boxplot(df[,IV] ~ df[,DV], col = color, main=paste("Boxplot of",DV,"by",IV), ylab=IV, xlab=D
V)
  }
  else if (class(df[,IV])=="factor") {
    t <- table(df[,DV],df[,IV])
    p <- prop.table(t,2) #if want a propotion table
    par(mfrow=c(1,2))
    barplot(t, beside=T, col=color, ylim = yrange(t), main=paste("Barplots for",DV,"by",IV), yla
b="Frequency", xlab=IV)
    legend("topleft", levels(df[,DV]), pch=15, col=color, bty="n")
    barplot(p, col=color, ylab = "Percent of Total", xlab=IV)
    par(mfrow=c(1,1))
  }
}
```

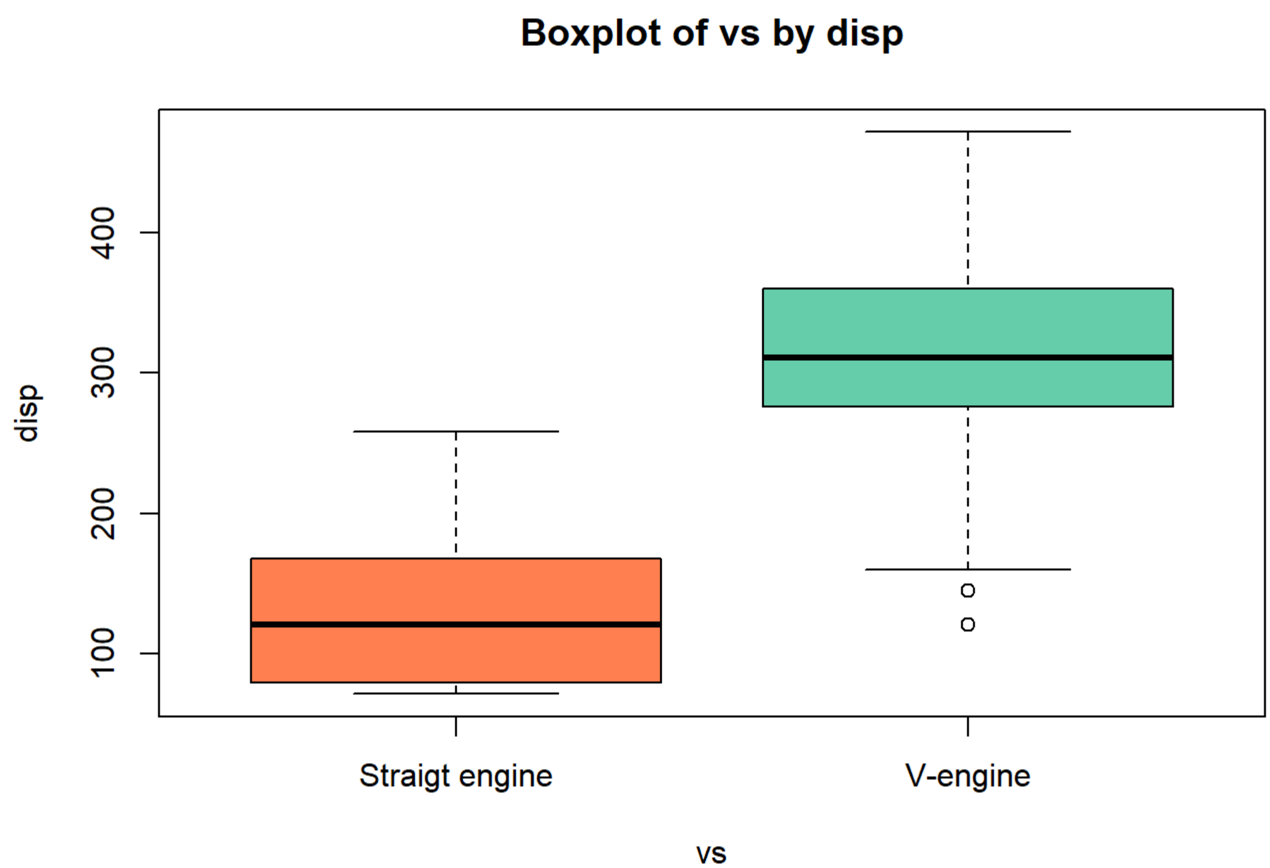
I execute the function by simply running the following.

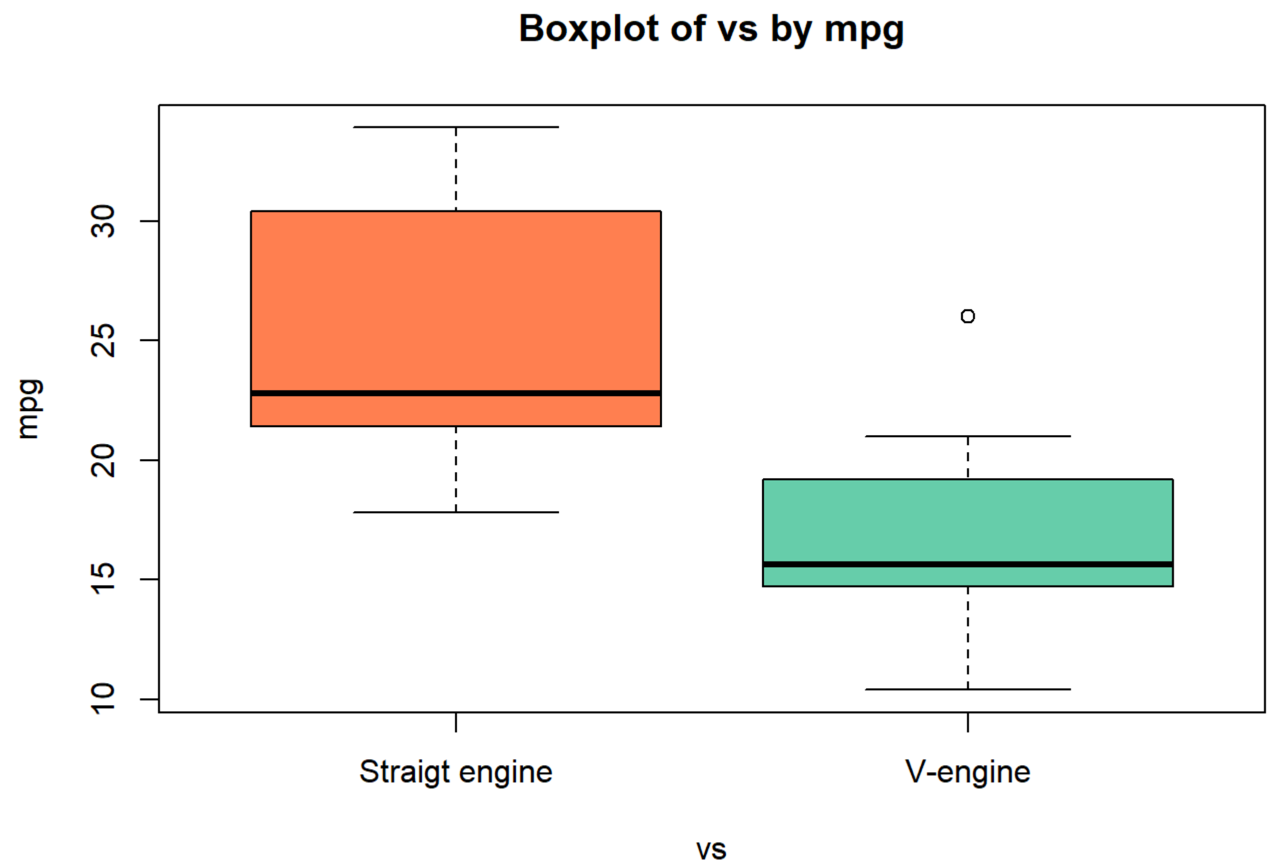
```
lapply(IVs, f.CatDvPlots)
```

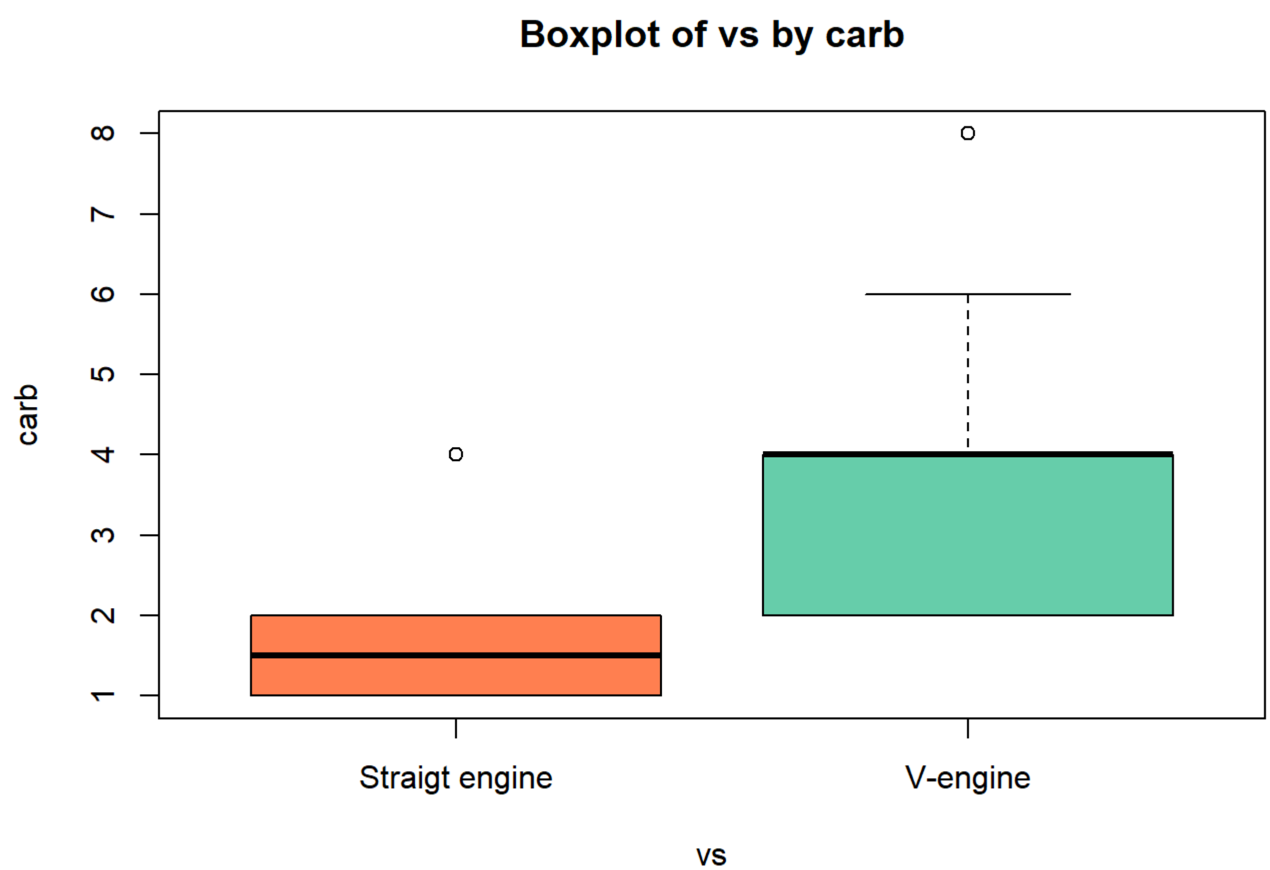

Boxplot of vs by qsec
qsec
22
20
18
16
Straigt engine
V-engine
vs

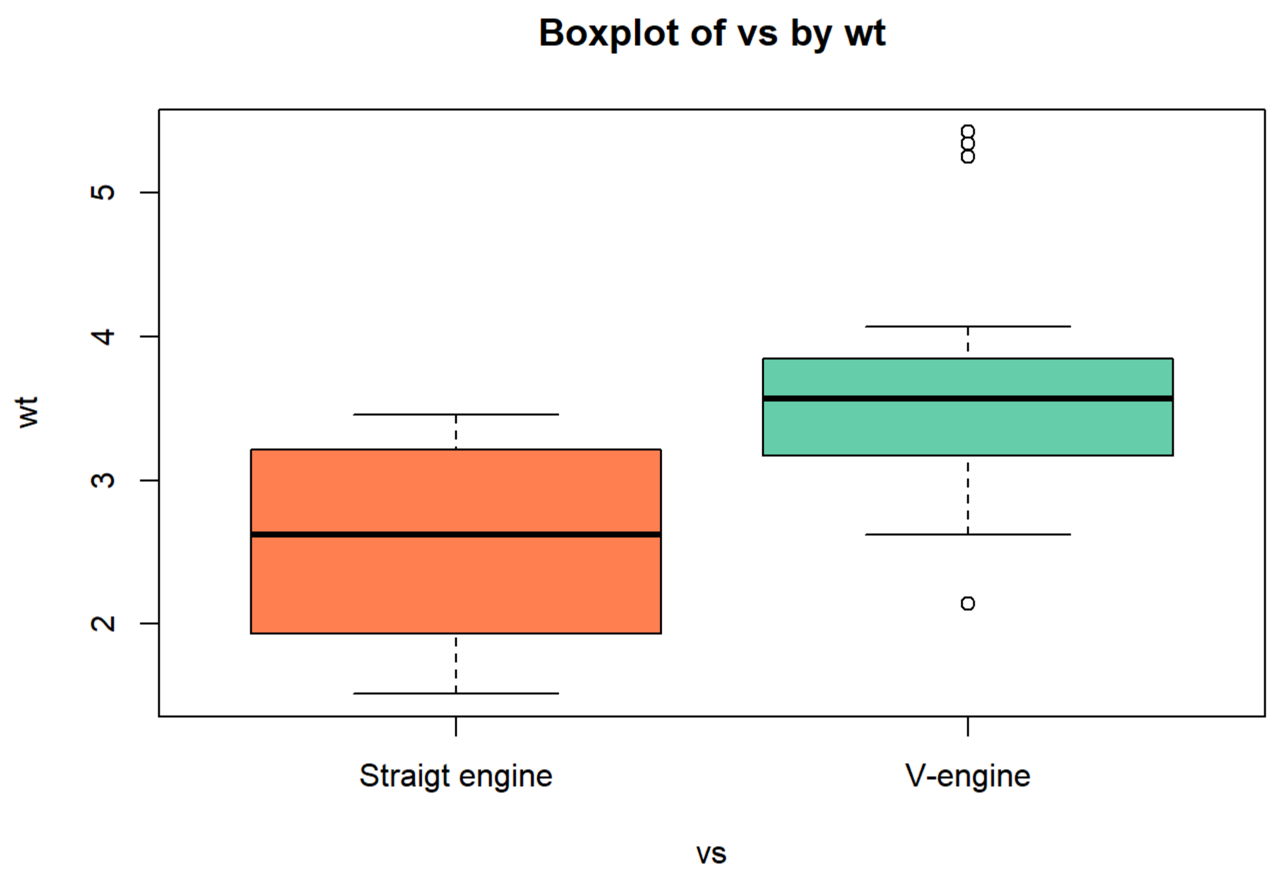

Barplots for vs by cyl
Straigt engine
V-engine
Frequency
15
10
5
0
4
6
8
cyl
Percent of Total
1.0
0.8
0.6
0.4
0.2
0.0
4
6
8
cyl

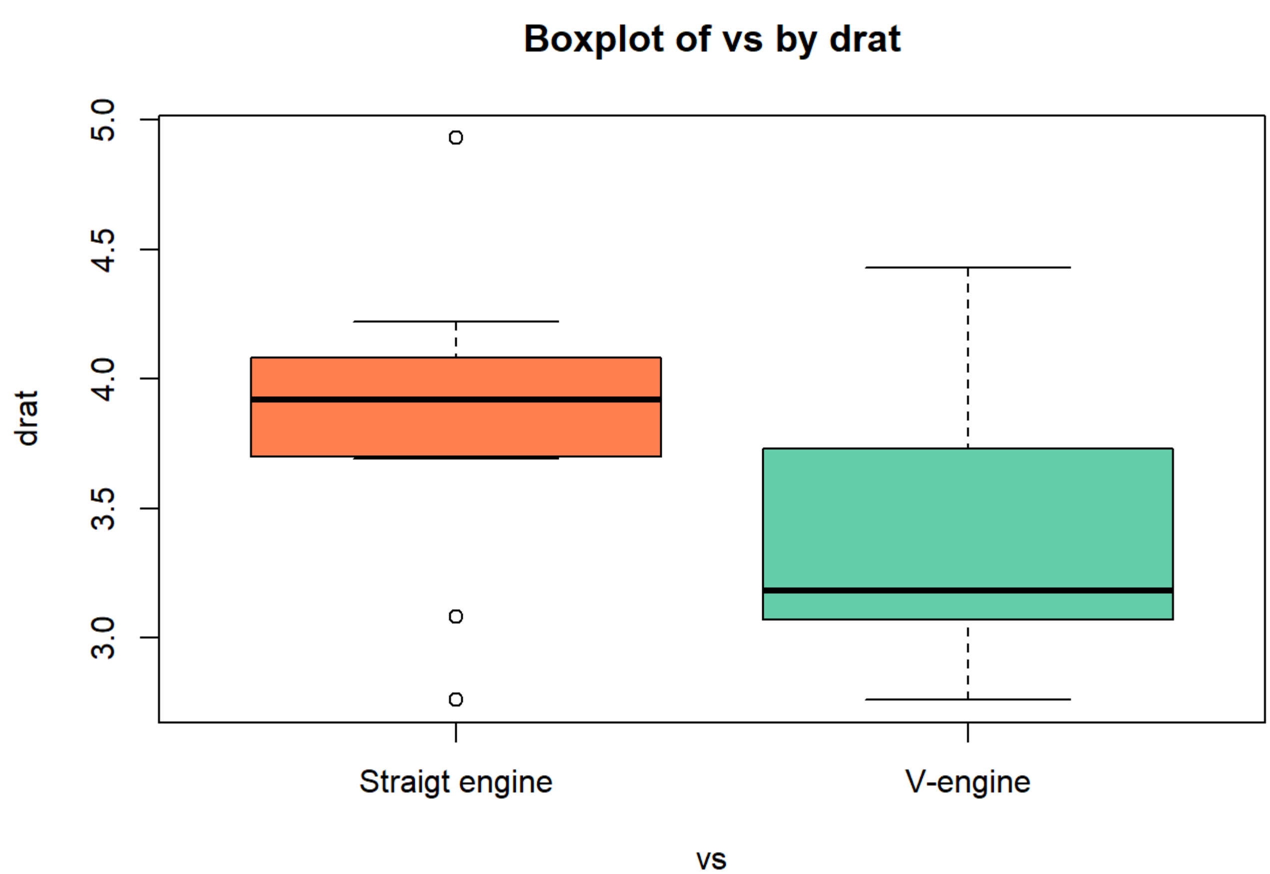

Boxplot of vs by hp
hp
50
100
150
200
250
300
Straigt engine
V-engine
vs


Boxplot of vs by disp
disp
100
200
300
400
Straigt engine
V-engine
vs


Boxplot of vs by mpg
mpg
10
15
20
25
30
Straigt engine
V-engine
vs


Boxplot of vs by carb
carb
1
2
3
4
5
6
7
8
Straigt engine
V-engine
vs


Boxplot of vs by wt
wt
5
4
3
2
Straigt engine
V-engine
vs


Boxplot of vs by drat
drat
5.0
4.5
4.0
3.5
3.0
Straigt engine
V-engine
vs