Regression Models Project

Hsin Chih Chen

10/13/2021

Project Introduction

This project assignment is to look at a data set of car collections in 1974 (**mtcars**), and the two main questions this project would like to know the following questions.

1. "Is automatic or manual transmission better for MPG"?
2. "Quantify the MPG difference between automatic and manual transmissions"

Preliminary Setup & Understanding the Data Structure.

Load the respective libraries for data obtainment, plotting statistical evaluation before executing the codes.

```
# load datasets to obtain the mtcars within the library.
library(datasets)

# load ggplot2 for the graphing algorithm.
library(ggplot2)

# load the car package to evaluate the VIFs for the mtcars model
library(car)
```

```
## Loading required package: carData
```

```
# load data table into the abbreviation
mtc <- mtcars

# Understand the data structure
sapply(mtc, class)
```

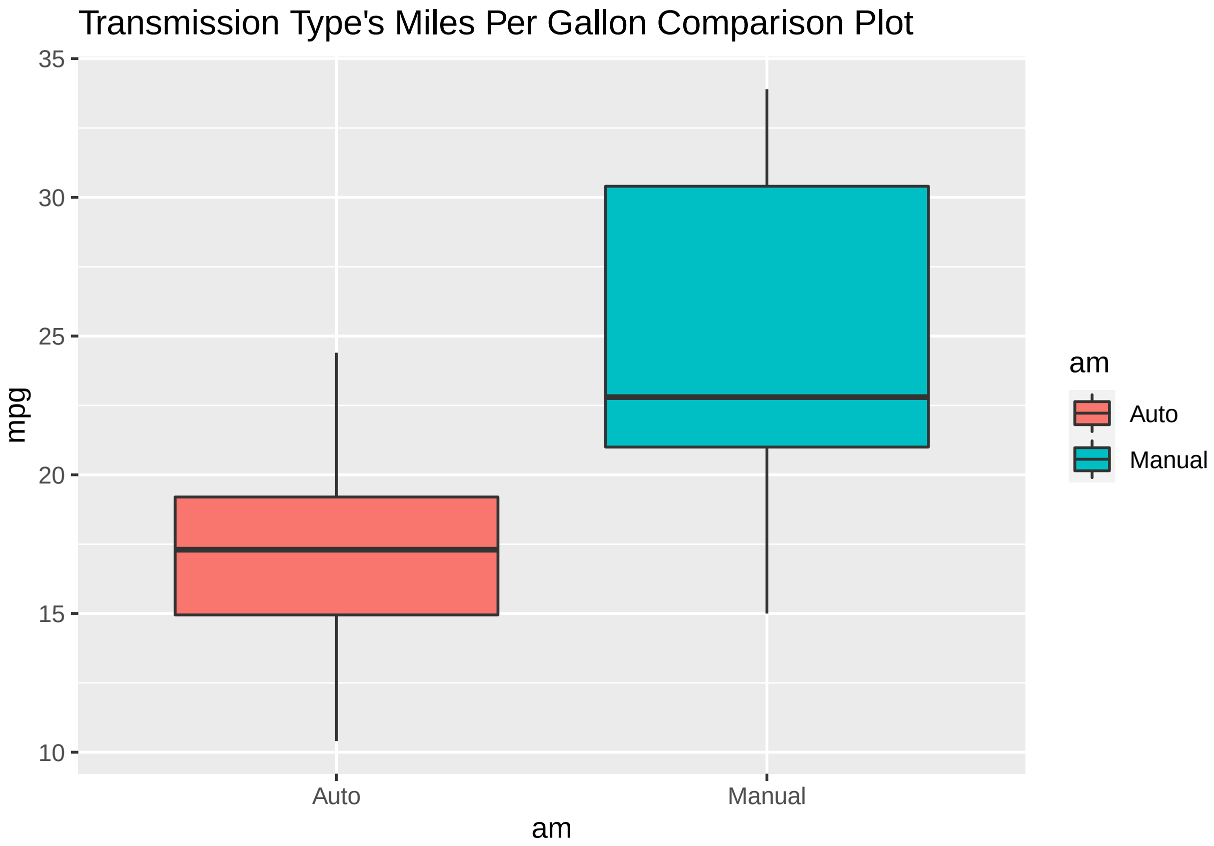
```
##       mpg       cyl      disp        hp      drat        wt      qsec        vs
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
##        am      gear      carb
## "numeric" "numeric" "numeric"
```

And by looking at the data structure, variable *am* is the transmission type (where 0 = automatic and 1 = manual from **?mtcars**'s justification). Therefore the model will definitely have this as an inclusion.

Default Model Setup.

In this particular model, we have used the factor labels to place the am as an identified parameter to indicate the 2 different population for comparison. The associated plot is attached with the code below:

```
# Use factor to label transmission
mtc$am <- factor(mtc$am, labels = c("Auto", "Manual"))

# Use ggplot to obtain the boxplot between the 2 transmission type
ggplot(data = mtc, aes(x = am, y = mpg)) +
geom_boxplot(aes(fill = am)) +
ggtitle("Transmission Type's Miles Per Gallon Comparison Plot")
```



```
# Create model for mpg and transmission type ONLY.
fit_t0 <- lm(mpg ~ am, data = mtc)
summary(fit_t0)$coef
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 17.147368   1.124603 15.247492 1.133983e-15
## amManual     7.244939   1.764422  4.106127 2.850207e-04
```

```
anova(fit_t0)
```

```
## Analysis of Variance Table
```

```
##
## Response: mpg
##           Df Sum Sq Mean Sq F value   Pr(>F)
## am         1 405.15  405.15   16.86 0.000285 ***
## Residuals 30 720.90   24.03
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By default, the ANOVA's hypothesis testing is given below.

H0 = Manual & Automatic will impact the mpg in the same way. H1 = Manual & Automatic will impact the mpg in the different way.

Based on the ANOVA summary and given box-plot, it indicates that the H0 should be rejected which indicates the contribution between transmission type certainly will impact the mpg output.

More Thorough Model Selection Approach

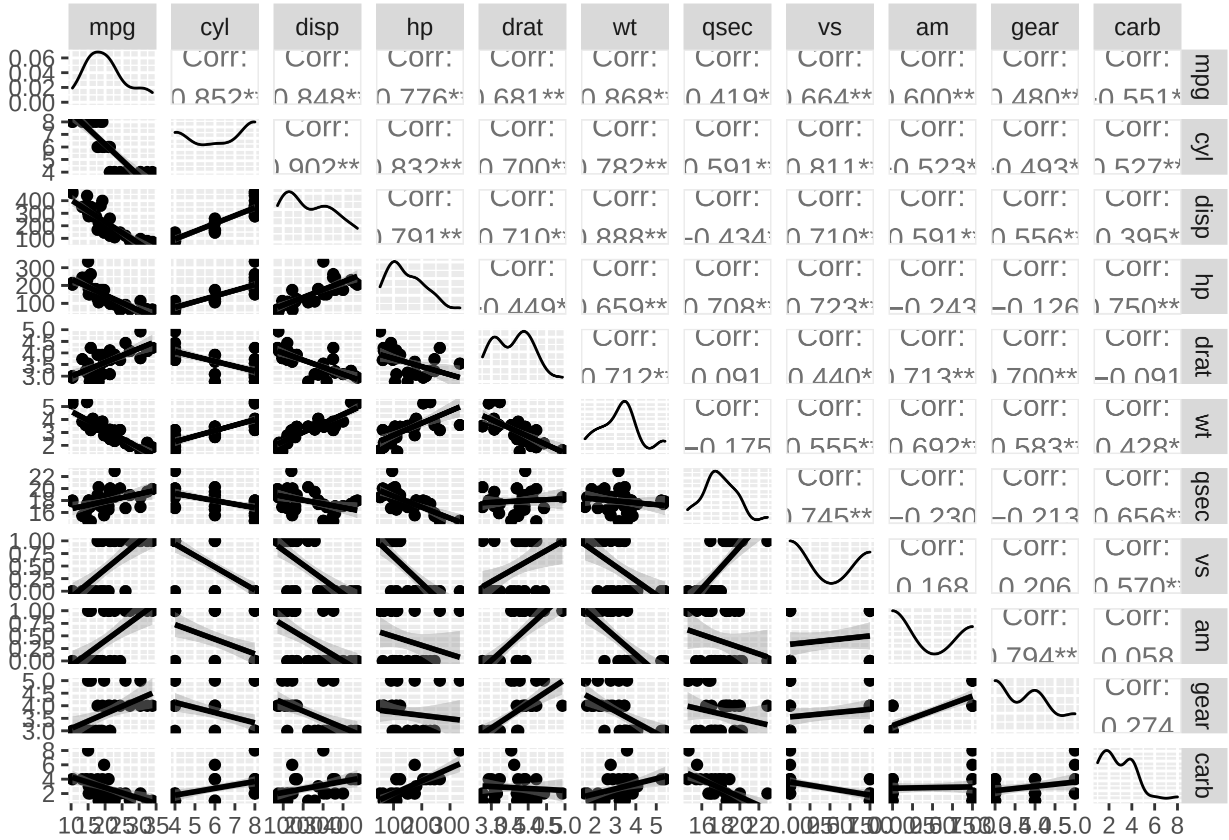
Correlation & VIF Confirmations

Before evaluating the collinearities between the input variable, finding the correlations between variables.

```
library(GGally)
```

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

```
cor_plot <- ggpairs(mtcars, lower = list(continuous = "smooth"))
cor_plot
```



In general, all parameters within the data frame shall be considered to evaluate the plotting. Therefore, this section will initiate the VIF examination while considering the collinearities within the parameters.

The VIF criteria is given below with the rule of thumb:

VIF	Status of Parameters
VIF = 1	Uncorrelated (and zero inflation)
1 < VIF < 5	Moderately correlated
VIF > 5 to 10	Highly correlated

Based on the VIF information for the entire data set (by setting mpg as the output variable), the VIF factors are given below.

```r
# Generate original setup for the full dataset model.
fit_full <- lm(mpg ~. , data = mtc)

# Check collinearity for full data set.
vif(fit_full)
```

```
##       cyl      disp        hp      drat        wt      qsec        vs        am
## 15.373833 21.620241  9.832037  3.374620 15.164887  7.527958  4.965873  4.648487
##      gear      carb
##  5.357452  7.908747
```

Based on observation, cyl and disp has high variance within the model, therefore removing the cylinder and disp shall reduce the VIF level for other factors.

```r
# Removing cyl and disp first for adjusted model
fit_t1 <- lm(mpg ~. -cyl -disp  , data = mtc)
summary(fit_t1)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp, data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8187 -1.3903 -0.3045  1.2269  4.5183
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.80810   12.88582   1.072   0.2950
## hp          -0.01225    0.01649  -0.743   0.4650
## drat         0.88894    1.52061   0.585   0.5645
## wt          -2.60968    1.15878  -2.252   0.0342 *
## qsec         0.63983    0.62752   1.020   0.3185
## vs           0.08786    1.88992   0.046   0.9633
## amManual     2.42418    1.91227   1.268   0.2176
## gear         0.69390    1.35294   0.513   0.6129
## carb        -0.61286    0.59109  -1.037   0.3106
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.566 on 23 degrees of freedom
## Multiple R-squared:  0.8655, Adjusted R-squared:  0.8187
## F-statistic:  18.5 on 8 and 23 DF,  p-value: 2.627e-08
```

```r
# Re-Check collinearity after removing the high variant factors
vif(fit_t1)
```

```
##       hp     drat       wt     qsec       vs       am     gear     carb
## 6.015788 3.111501 6.051127 5.918682 4.270956 4.285815 4.690187 4.290468
```

Model Elimination & Fitting

After remove the top 2 confounding factors which caused the high collinearity, elimination of additional factors which does not have correlation with respect to the mpg output will be slowly eliminated based on the p-value of 95% confidence level.

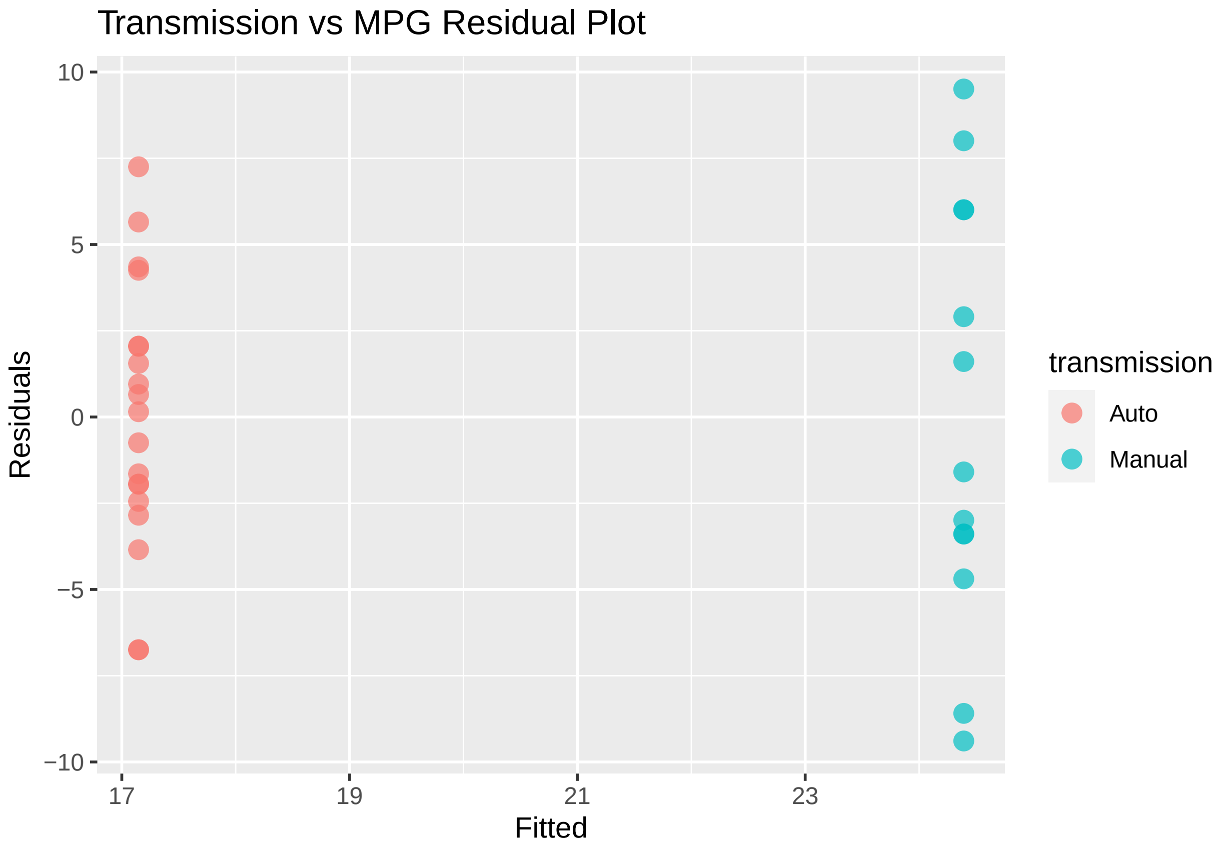
```r
# Remove cyl, disp and vs
fit_t2 <- lm(mpg ~. -cyl -disp -vs, data = mtc)

# Remove cyl, disp, vs and gear
fit_t3 <- lm(mpg ~. -cyl -disp -vs -gear, data = mtc)

# Remove cyl, disp, vs ,gear and hp
fit_t4 <- lm(mpg ~. -cyl -disp -vs -gear -hp, data = mtc)
```

```
# Remove cyl, vs, carb , gear, hp and drat
fit_t5 <- lm(mpg ~. -cyl -disp -vs -gear -hp -drat, data = mtc)

# Remove cyl, vs, carb, gear, hp, drat and carb
fit_t6 <- lm(mpg ~. -cyl -disp -vs -gear -hp -drat -carb, data = mtc)
```

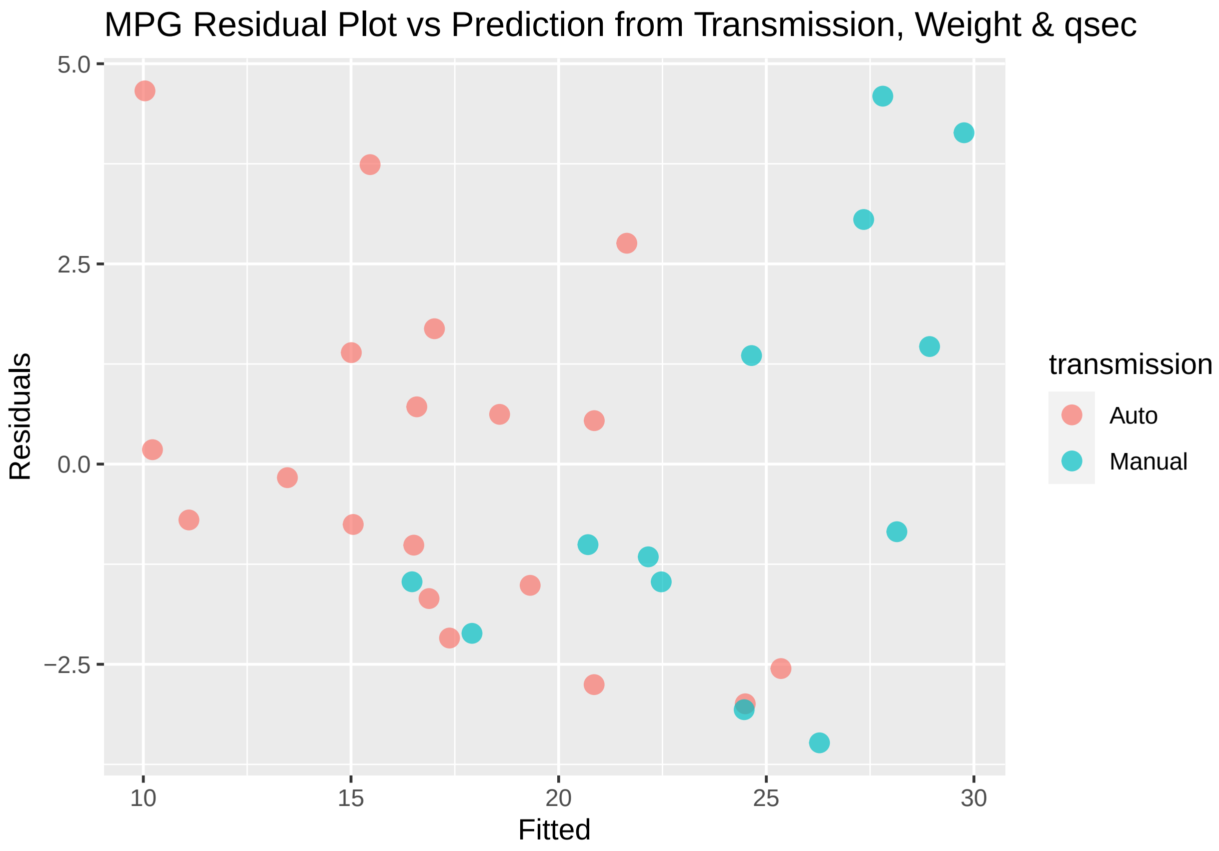
Residual and Inference Analysis

In this section, the linear model for mpg vs transmission type and the ideal model based on prediction from **transmission type**, **weight** and **qsec** will be applied for the calculation values will plotting the regression line.

```
# setup the residuals and predicted values for plotting based on the linear model by transmission vs mp

par(mfrow = c(2,2))
fit_t0s <- data.frame(Fitted = predict(fit_t0), Residuals = resid(fit_t0), transmission = mtc$am)

fit_t6s <- data.frame(Fitted = predict(fit_t6), Residuals = resid(fit_t6), transmission = mtc$am)

ggplot(data = fit_t0s, aes(x = Fitted, y = Residuals, color = transmission)) +
geom_point(size = 3, alpha = 0.7)+
  ggtitle("Transmission vs MPG Residual Plot")
```



```r
ggplot(data = fit_t6s, aes(x = Fitted, y = Residuals, color = transmission)) +
geom_point(size = 3, alpha = 0.7)+
  ggtitle("MPG Residual Plot vs Prediction from Transmission, Weight & qsec")
```



The following codes are the respective factors within the model.

```r
# Change in outcome
dffits(fit_t0)
dffits(fit_t6)

# Change in individual coefficients
dfbetas(fit_t0)
dfbetas(fit_t6)

# Overall change in coefficients
cooks.distance(fit_t0)
cooks.distance(fit_t6)

# Leverage Comparison
hatvalues(fit_t0)
hatvalues(fit_t6)
```

Conclusion

Based on the given information, the following conclusion can be made.

1. The transmission type **does** effect the mpg differently and is considered as a two separate population within the mtcars data set. The residual plots and p-value validates this hypothesis
2. After consolidating the correlation factors, the best way to predict the miles per gallon besides transmission type are the **wt** and **qsec** variables.
3. The confidence interval for both automatic transmission & predictive value range is given below.

Type	Mean	Lower	Upper
Automatic (Confidence Interval)	17.15	14.85	19.44
Automatic (Prediction)	17.15	6.88	27.42
Manual (Confidence Interval)	24.39	21.62	27.17
Manual (Prediction)	24.39	14.00	34.78

Appendix

```r
# Summarize all fit models respectively
summary(fit_t2)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp - vs, data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7967 -1.4077 -0.2955  1.2099  4.5072
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.49215   10.71794   1.259   0.2202
## hp          -0.01215    0.01599  -0.760   0.4549
## drat         0.89764    1.47732   0.608   0.5491
## wt          -2.62772    1.06891  -2.458   0.0216 *
## qsec         0.65845    0.47292   1.392   0.1766
## amManual     2.41351    1.85858   1.299   0.2064
## gear         0.70547    1.30189   0.542   0.5929
## carb        -0.61460    0.57750  -1.064   0.2978
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.512 on 24 degrees of freedom
## Multiple R-squared:  0.8655, Adjusted R-squared:  0.8262
## F-statistic: 22.06 on 7 and 24 DF,  p-value: 5.308e-09
```

```
summary(fit_t3)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp - vs - gear, data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7903 -1.3426 -0.1935  1.1624  4.2998
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 14.55110   10.38833   1.401  0.17359
## hp          -0.01174    0.01575  -0.746  0.46292
## drat         1.07285    1.42100   0.755  0.45731
## wt          -2.82895    0.98807  -2.863  0.00837 **
## qsec         0.70712    0.45770   1.545  0.13493
## amManual     2.85861    1.64350   1.739  0.09427 .
## carb        -0.45445    0.48908  -0.929  0.36168
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.477 on 25 degrees of freedom
## Multiple R-squared:  0.8638, Adjusted R-squared:  0.8311
## F-statistic: 26.43 on 6 and 25 DF,  p-value: 1.122e-09
```

```
summary(fit_t4)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp - vs - gear - hp, data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9355 -1.2134 -0.3151  1.0669  4.2271
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.9243     8.2592   1.202  0.24035
## drat          1.2071     1.3975   0.864  0.39562
## wt           -3.1108     0.9050  -3.437  0.00199 **
## qsec          0.9145     0.3603   2.538  0.01748 *
## amManual      2.9639     1.6234   1.826  0.07939 .
## carb         -0.6023     0.4432  -1.359  0.18583
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.455 on 26 degrees of freedom
## Multiple R-squared:  0.8608, Adjusted R-squared:  0.834
## F-statistic: 32.15 on 5 and 26 DF,  p-value: 2.423e-10
```

```
summary(fit_t5)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp - vs - gear - hp - drat, data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1184 -1.5414 -0.1392  1.2917  4.3604
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  12.8972     7.4725   1.726 0.095784 .
## wt           -3.4343     0.8200  -4.188 0.000269 ***
## qsec          1.0191     0.3378   3.017 0.005507 **
## amManual      3.5114     1.4875   2.361 0.025721 *
## carb         -0.4886     0.4212  -1.160 0.256212
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.444 on 27 degrees of freedom
## Multiple R-squared:  0.8568, Adjusted R-squared:  0.8356
## F-statistic: 40.39 on 4 and 27 DF,  p-value: 5.064e-11
```

```
summary(fit_t6)
```

```
##
## Call:
## lm(formula = mpg ~ . - cyl - disp - vs - gear - hp - drat - carb,
##     data = mtc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4811 -1.5555 -0.7257  1.4110  4.6610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9.6178     6.9596   1.382 0.177915
## wt           -3.9165     0.7112  -5.507 6.95e-06 ***
## qsec          1.2259     0.2887   4.247 0.000216 ***
## amManual      2.9358     1.4109   2.081 0.046716 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.459 on 28 degrees of freedom
## Multiple R-squared:  0.8497, Adjusted R-squared:  0.8336
## F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

The inference analysis is calculated below without the intercepts.

```
# Store the coefficient summary without intercepts
fit_t0t <- lm(mpg ~ am - 1, data = mtcars)
fit_t0c <- summary(fit_t0t)$coef

# Use Prediction and confidence level to evaluate the models

predict(fit_t0, newdata = data.frame(am = as.factor(c("Auto", "Manual"))), interval = "confidence")
```

```
##        fit      lwr      upr
## 1 17.14737 14.85062 19.44411
## 2 24.39231 21.61568 27.16894
```

```
predict(fit_t0, newdata = data.frame(am = as.factor(c("Auto", "Manual"))), interval = "prediction")
```

```
##        fit       lwr      upr
## 1 17.14737  6.876013 27.41872
## 2 24.39231 14.003113 34.78150
```