Assignment 5: Multiple linear regression; putting it all together

Matthew D. Sherman

STAT202: Regression Modelling

September 14, 2023

Question One:

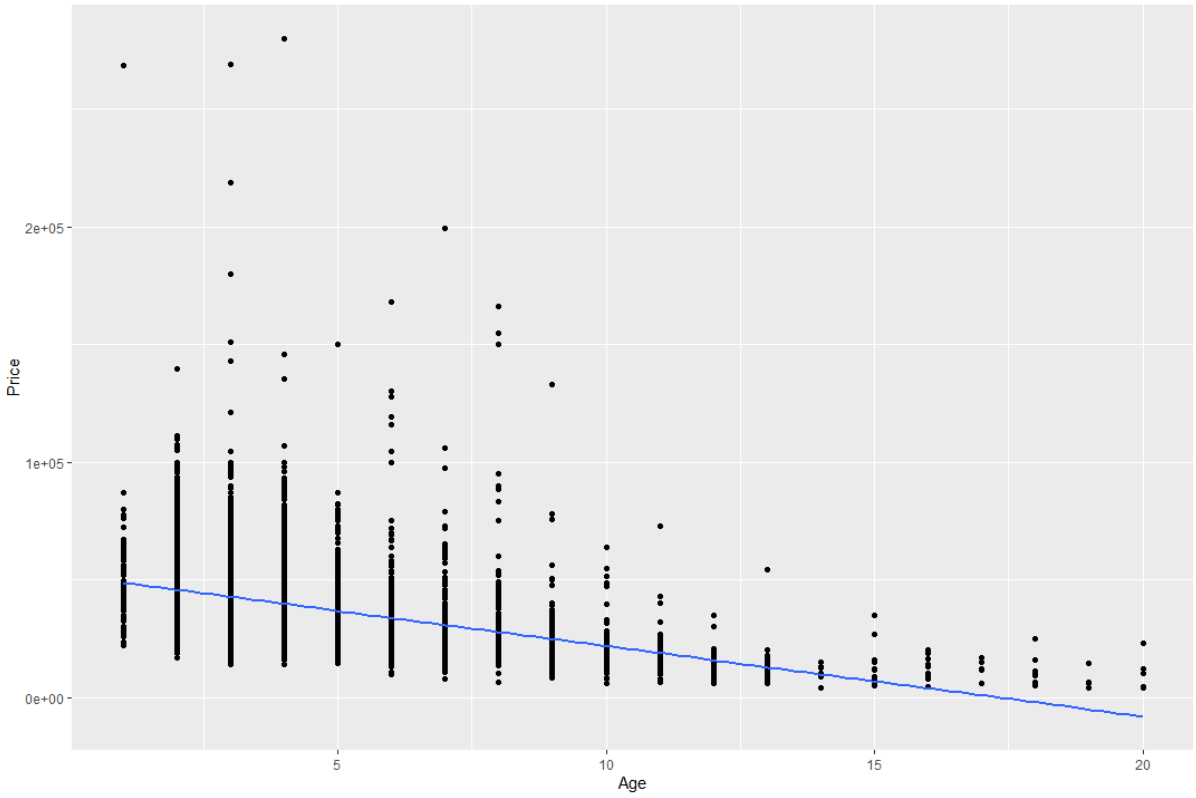
```
cars <- read_csv("cars_sales.zip")
```

Question Two:

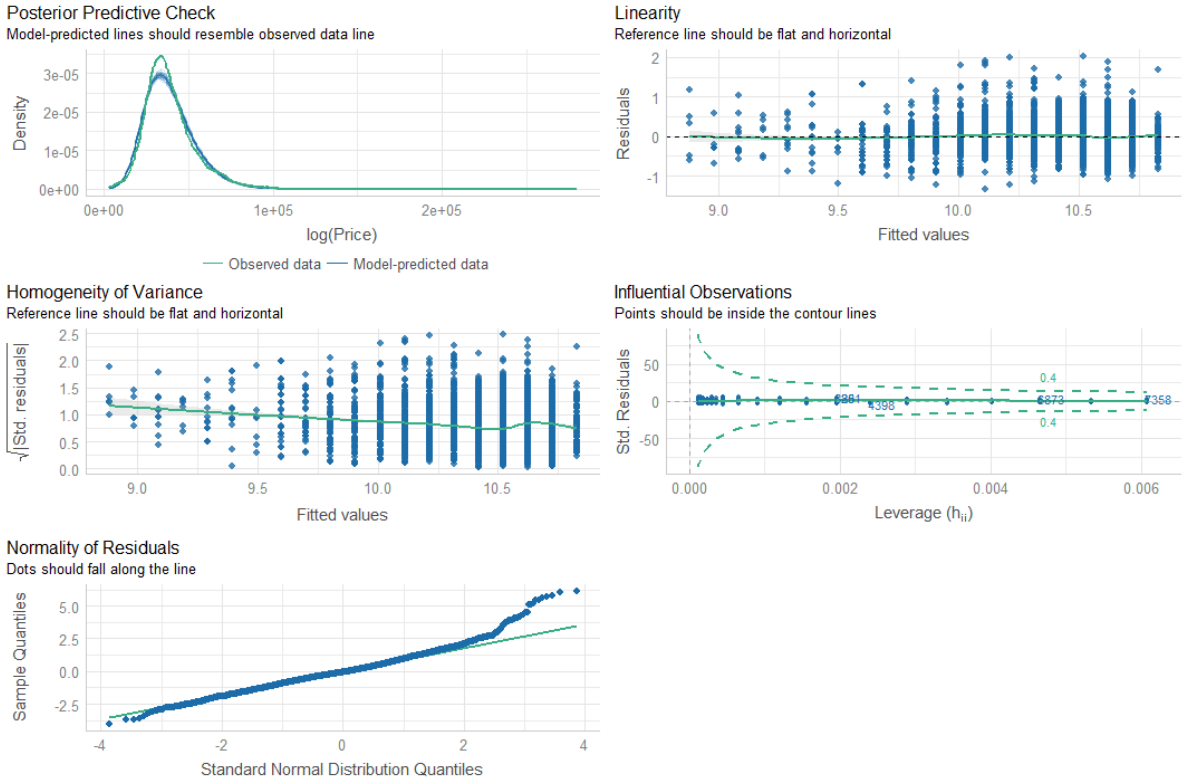
```
cars <- cars %>% filter(Price != "Not Priced")
cars <- cars %>% filter(!is.na(DealType))
cars <- cars %>% rename(UsedNew = `Used/New`)
cars <- cars %>% mutate(Price = parse_number(Price))
cars <- cars %>% mutate(SellerType = factor(SellerType))
cars <- cars %>% mutate(DealType = factor(DealType))
cars <- cars %>% mutate(Age = 2023 - Year)
set.seed(16645573)
my_cars <- cars %>% sample_n(9100)
```

Question Three:

```
ggplot(my_cars, aes(x = Age, y = Price)) + geom_point() +
geom_smooth(method=lm, se = FALSE)
```



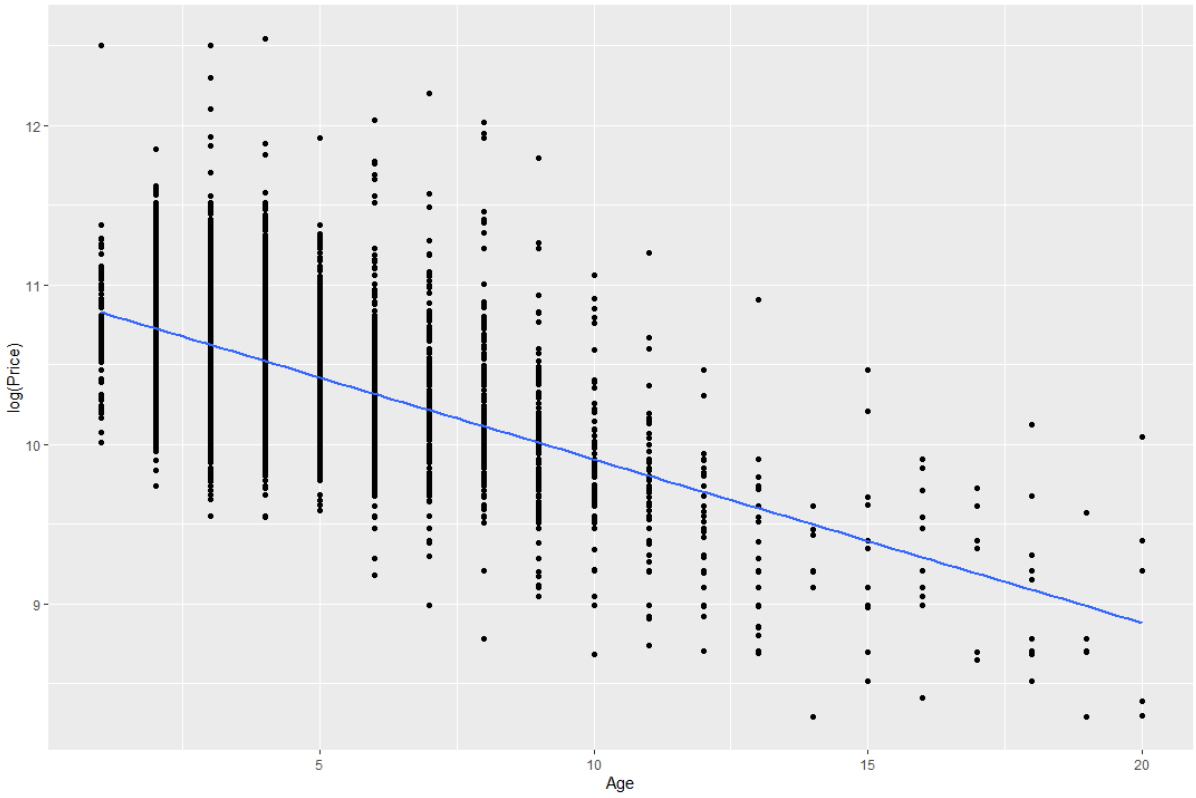
```
m1 <- lm(Price ~ Age, data = my_cars)

check_model(m1)
```



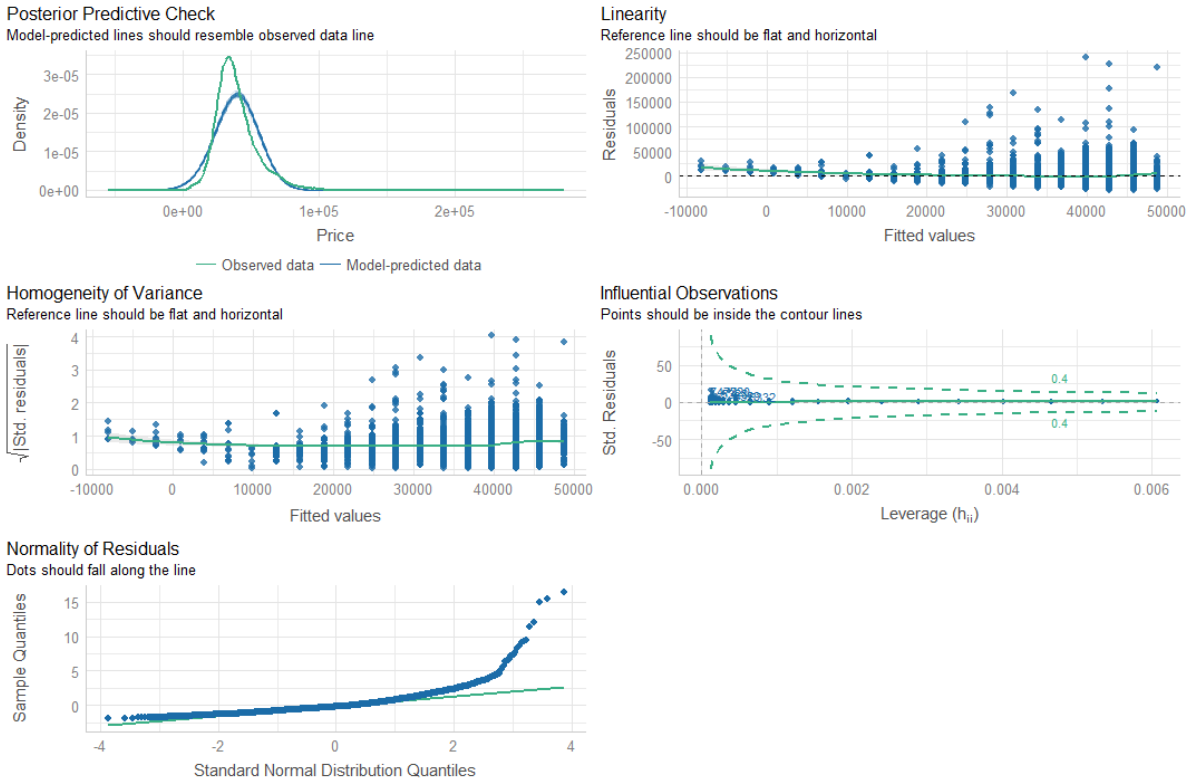
The residuals vs fitted plot shows linearity (and does not deviate from linearity), as indicated by the flat and horizontal reference line. There are no highly influential observations, as indicated by all the points being inside the contour lines in the influential observations (residuals vs leverage) plot. There is some deviation from normality towards the right tail end of the normality of residuals plot. The homogeneity of variance plot shows a reference line which, while on a slight downwards diagonal, is overall satisfactory.

Question Four:

```
ggplot(my_cars, aes(x = Age, y = log(Price))) + geom_point() +

geom_smooth(method=lm, se = FALSE)
```




```
m2 <- lm(log(Price) ~ Age, data = my_cars)

check_model(m2)
```



The residuals vs fitted values show a reference line is flat and horizontal, indicating linearity and a lack of deviations from linearity. The influential observations plot (residuals vs leverage) shows no points outside the contour lines, indicating that there are no highly influential observations. The homogeneity of variance plot shows a reference line which is very flat and horizontal, indicating constant variance. The normality of residuals plot shows a somewhat substantial deviation from normality towards the right tail end.

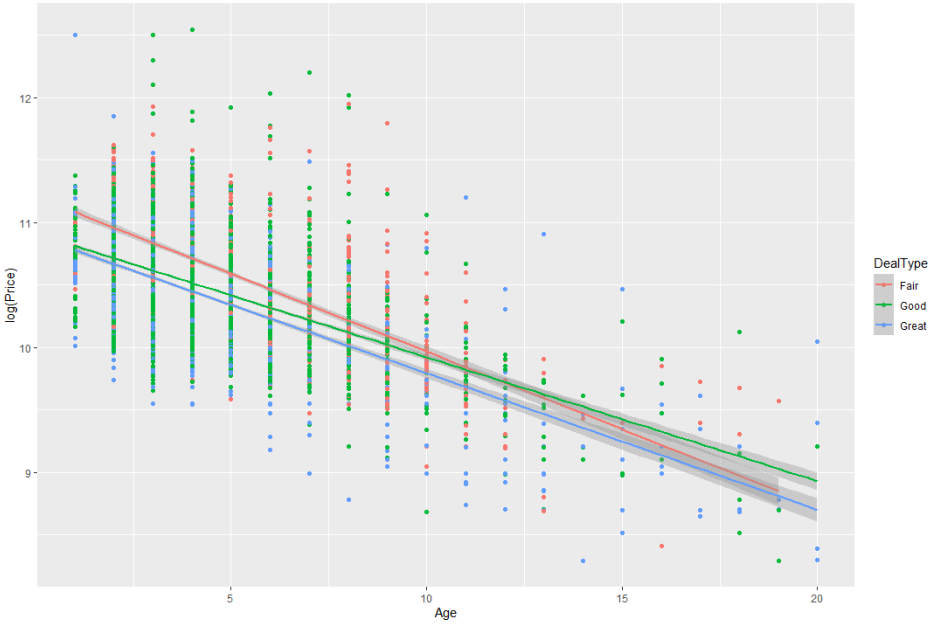
Question Five:

```
ggplot(data = my_cars, aes(x = Age, y = log(Price), color = DealType)) +
geom_point()
```

Before fitting other models, how likely do you think that the regression lines differ between Deal Types?(20 words)

I think it's very likely due to the variation in data points between the deal types as shown in the scatter plot.

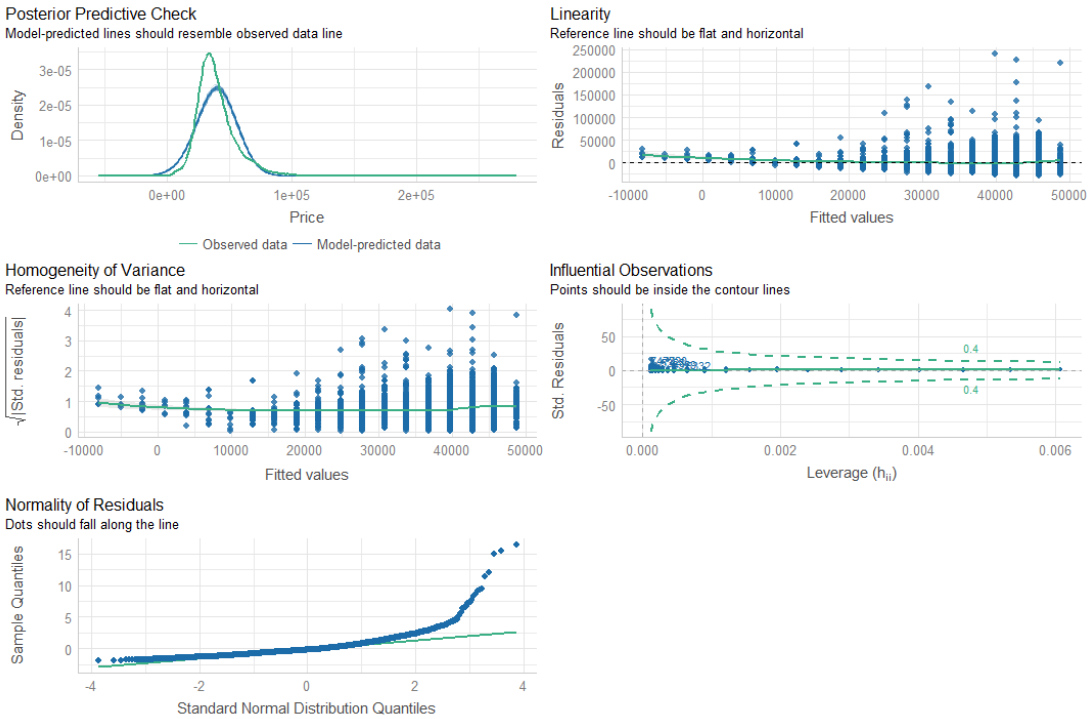
```
ggplot(data = my_cars, aes(x = Age, y = log(Price), color = DealType)) +
geom_point() + geom_smooth(method=lm)
```



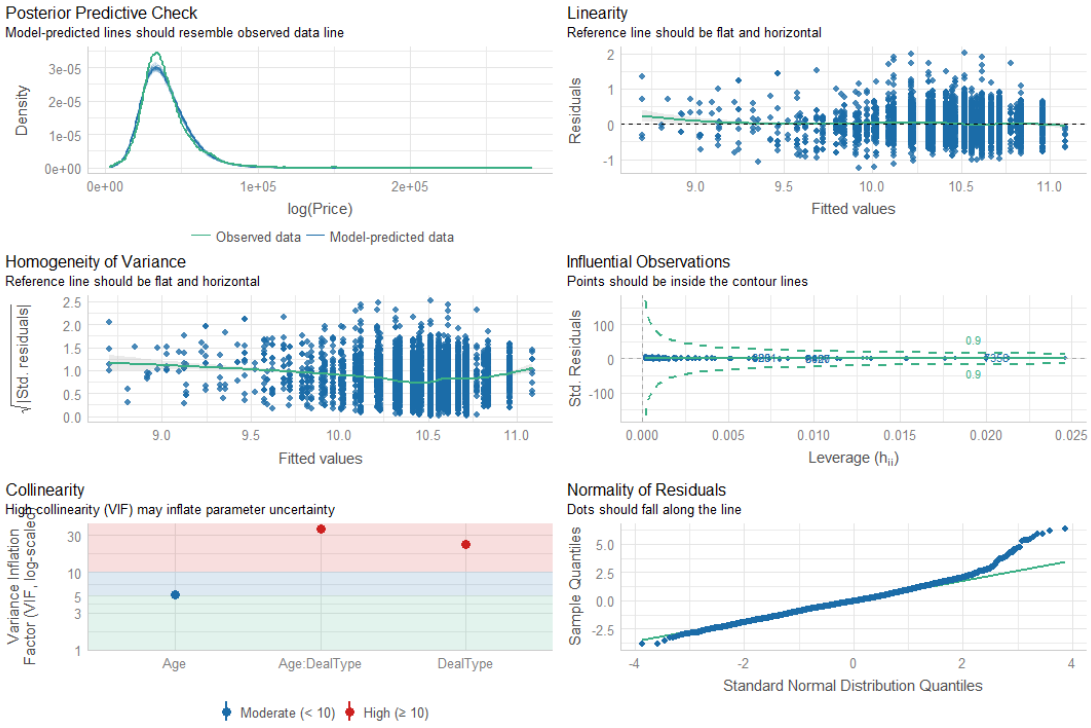
Question Six:

```
m3 <- lm(log(Price) ~ Age * DealType, data = my_cars)
check_model(m3)
```

m2:



m3:



The residuals vs fitted values plots between the two models are similar with a very small deviation from linearity at the left end of the reference line in both, but both are very satisfactory in terms of linearity. Boths plots have no highly influential observations as shown by all the points being between the contour lines in the influential observations (residuals vs leverage) plots. The homogeneity of variance plot in m2 shows slightly more favourable constant variance compared to m3, but both are satisfactory in terms of constant variance. The normality of residuals plots in both models show a deviation in normality towards the right tail, however, m3 is more favourable in terms of normality compared to m2 as the deviation at its right tail is considerably less severe.

Question Seven:

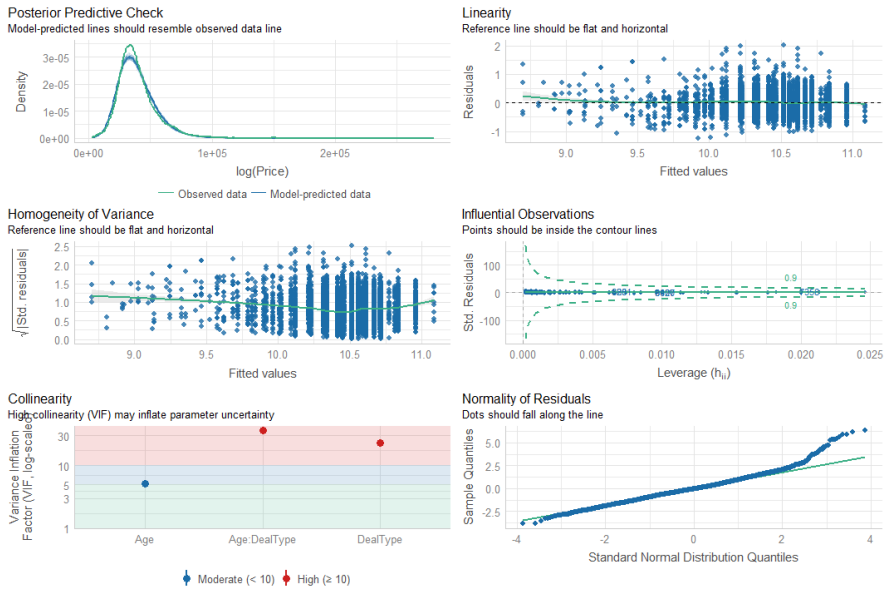
```
m4 <- lm(log(Price) ~ Age * DealType + Mileage, data = my_cars)
summary(m3)
summary(m4)
check_model(m4)
```

```
> summary(m3)

Call:
lm(formula = log(Price) ~ Age * DealType, data = my_cars)

Residuals:
     Min       1Q   Median       3Q      Max
-1.23768 -0.20042 -0.01685  0.19103  2.02679

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       11.209559   0.020492 547.027  < 2e-16 ***
Age               -0.124333   0.003576 -34.766  < 2e-16 ***
DealTypeGood      -0.297187   0.022956 -12.946  < 2e-16 ***
DealTypeGreat     -0.323052   0.024540 -13.164  < 2e-16 ***
Age:DealTypeGood   0.025135   0.004251   5.913 3.48e-09 ***
Age:DealTypeGreat  0.014968   0.004568   3.277  0.00105 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3219 on 9094 degrees of freedom
Multiple R-squared:  0.3417,    Adjusted R-squared:  0.3414
F-statistic: 944.3 on 5 and 9094 DF,  p-value: < 2.2e-16
```

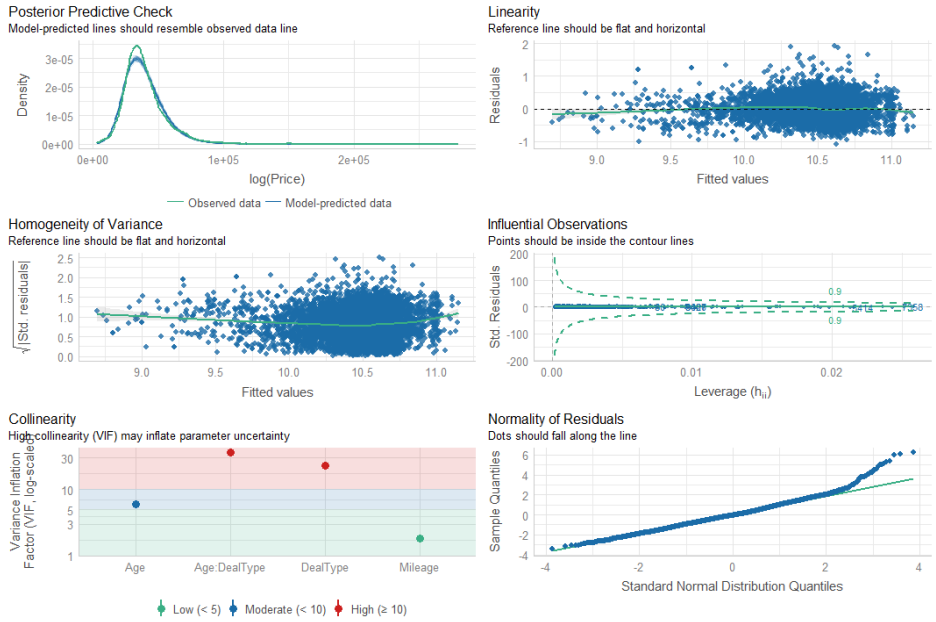
```
> summary(m4)

Call:
lm(formula = log(Price) ~ Age * DealType + Mileage, data = my_cars)

Residuals:
     Min       1Q   Median       3Q      Max
-1.07551 -0.20097 -0.01393  0.19017  1.93084

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       1.124e+01  1.986e-02 566.238  < 2e-16 ***
Age              -9.014e-02  3.714e-03 -24.271  < 2e-16 ***
DealTypeGood     -3.161e-01  2.221e-02 -14.235  < 2e-16 ***
DealTypeGreat    -3.241e-01  2.373e-02 -13.662  < 2e-16 ***
Mileage          -4.527e-06  1.794e-07 -25.230  < 2e-16 ***
Age:DealTypeGood  2.714e-02  4.110e-03   6.602 4.29e-11 ***
Age:DealTypeGreat 1.217e-02  4.418e-03   2.755  0.00589 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3112 on 9093 degrees of freedom
Multiple R-squared:  0.3848,    Adjusted R-squared:  0.3844
F-statistic:   948 on 6 and 9093 DF,  p-value: < 2.2e-16
```

m4 has a better fit than m3 as its RSE is lower (indicating less residual standard error) and its multiple R-squared and adjusted R-squared values are higher (indicating better goodness of fit).

m3:



m4:



The residuals in m4 appear to be very similar compared to m3 in meeting the residual assumptions. Both linearity plots of the models show a flat and horizontal reference line, indicating linearity in the models. Both models also have no highly influential observations as indicated by neither residuals vs leverage plots have no points outside the contour lines. Both normality of residuals plots show a similar deviation from normality towards the right tail, however, the very small amount of deviation at the left tail for both NoR plots is slightly better in terms of normality for m4. The homogeneity of variance plots from both models shows a very similar reference line, however, the reference line for m4 appears to be very slightly flatter, indicating better constant variance. The collinearity plot in m4 appears to show age having a slightly higher VIF, while Age: DealType and DealType appear to have the same VIF for both models (and of course m4 has the extra mileage variable). Overall, there does not appear to be a substantial improvement in residual assumptions in m4 compared to m3, however, the model fit is improved so it is overall still a better model.

Question Eight:

```
coef(m4)
```

```
> coef(m4)
      (Intercept)               Age      DealTypeGood      DealTypeGreat           Mileage  Age:DealTypeGood
     1.124311e+01     -9.013600e-02     -3.160966e-01     -3.241394e-01     -4.526691e-06      2.713505e-02
Age:DealTypeGreat
     1.216977e-02
```

Regression line coefficients:

DealTypeGood: -0.3161

DealTypeGreat: -0.3241

DealTypeFair: Not provided as it is used as the reference.

Question Nine:

```
new_car <- tibble(Age = 2023 - 2017, DealType = "Great", Mileage = 55000)
exp(predict(m4, new_car))
exp(predict(m4, new_car, interval = "confidence"))
```

```
> exp(predict(m4, new_car))
       1
26962.48
> exp(predict(m4, new_car, interval = "confidence"))
       fit      lwr      upr
1 26962.48 26530.83 27401.15
```

- Predicted value (fit): 26962.48
- Lower bound of confidence interval (lwr): 26530.83
- Upper bound of confidence interval (upr): 27401.15

Question Ten:

```
m5 <- lm(log(Price) ~ Age * DealType + Mileage + ConsumerRating, data =
my_cars)
summary(m4)
summary(m5)
```

```
> summary(m4)

Call:
lm(formula = log(Price) ~ Age * DealType + Mileage, data = my_cars)

Residuals:
     Min       1Q   Median       3Q      Max
-1.07551 -0.20097 -0.01393  0.19017  1.93084

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.124e+01  1.986e-02 566.238  < 2e-16 ***
Age               -9.014e-02  3.714e-03 -24.271  < 2e-16 ***
DealTypeGood      -3.161e-01  2.221e-02 -14.235  < 2e-16 ***
DealTypeGreat     -3.241e-01  2.373e-02 -13.662  < 2e-16 ***
Mileage           -4.527e-06  1.794e-07 -25.230  < 2e-16 ***
Age:DealTypeGood   2.714e-02  4.110e-03   6.602 4.29e-11 ***
Age:DealTypeGreat  1.217e-02  4.418e-03   2.755  0.00589 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3112 on 9093 degrees of freedom
Multiple R-squared:  0.3848,    Adjusted R-squared:  0.3844
F-statistic:   948 on 6 and 9093 DF,  p-value: < 2.2e-16
```

```
> summary(m5)

Call:
lm(formula = log(Price) ~ Age * DealType + Mileage + ConsumerRating,
    data = my_cars)

Residuals:
     Min       1Q   Median       3Q      Max
-1.07753 -0.19759 -0.01425  0.19261  1.95995

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.196e+01  6.870e-02 174.037  < 2e-16 ***
Age               -9.121e-02  3.691e-03 -24.709  < 2e-16 ***
DealTypeGood      -3.149e-01  2.207e-02 -14.272  < 2e-16 ***
DealTypeGreat     -3.181e-01  2.358e-02 -13.490  < 2e-16 ***
Mileage           -4.499e-06  1.783e-07 -25.231  < 2e-16 ***
ConsumerRating    -1.510e-01  1.392e-02 -10.849  < 2e-16 ***
Age:DealTypeGood   2.699e-02  4.084e-03   6.607 4.13e-11 ***
Age:DealTypeGreat  1.031e-02  4.393e-03   2.347    0.019 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3092 on 9092 degrees of freedom
Multiple R-squared:  0.3927,    Adjusted R-squared:  0.3922
F-statistic: 839.8 on 7 and 9092 DF,  p-value: < 2.2e-16
```

m5 is a better model compared to m4 as it has a lower RSE (indicating less residual standard error) and higher multiple R-squared and Adjusted R-squared values compared to m4 (indicating better goodness-of-fit), overall indicating that m5 has a better fit over m4.

Full Code:

```
library(tidyverse)
library(performance)


cars <- read_csv("cars_sales.zip")
```

```
cars <- cars %>% filter(Price != "Not Priced")
cars <- cars %>% filter(!is.na(DealType))
cars <- cars %>% rename(UsedNew = `Used/New`)
cars <- cars %>% mutate(Price = parse_number(Price))
cars <- cars %>% mutate(SellerType = factor(SellerType))
cars <- cars %>% mutate(DealType = factor(DealType))
cars <- cars %>% mutate(Age = 2023 - Year)
set.seed(16645573)
my_cars <- cars %>% sample_n(9100)

ggplot(my_cars, aes(x = Age, y = Price)) + geom_point() + geom_smooth(method=lm, se =
FALSE)
m1 <- lm(Price ~ Age, data = my_cars)
check_model(m1)

ggplot(my_cars, aes(x = Age, y = log(Price))) + geom_point() + geom_smooth(method=lm, se =
FALSE)
m2 <- lm(log(Price) ~ Age, data = my_cars)
check_model(m2)

ggplot(data = my_cars, aes(x = Age, y = log(Price), color = DealType)) + geom_point()
ggplot(data = my_cars, aes(x = Age, y = log(Price), color = DealType)) + geom_point() +
geom_smooth(method=lm)

m3 <- lm(log(Price) ~ Age * DealType, data = my_cars)
check_model(m3)

m4 <- lm(log(Price) ~ Age * DealType + Mileage, data = my_cars)
summary(m3)
summary(m4)
check_model(m4)

coef(m4)

new_car <- tibble(Age = 2023 - 2017, DealType = "Great", Mileage = 55000)
exp(predict(m4, new_car))
exp(predict(m4, new_car, interval = "confidence"))

m5 <- lm(log(Price) ~ Age * DealType + Mileage + ConsumerRating, data = my_cars)
summary(m4)
summary(m5)
```