Homework 8-9-10

## 0) Data

This regression analysis consists of two different samples of 200 randomly selected observations from the nycflights13 dataset, filtered to include only flights operated by American Airlines (AA), United Airlines (UA), and Envoy Air (MQ). The 1st analysis includes the variables: arr\_delay (arrival delay), dep\_delay (departure delay), air\_time (duration of flight in minutes), and carrier (airline). The 2nd analysis includes the variables: dep\_delay, air\_time, distance, hour (hour of the day), and day (day of the month).

The goal of the 1st analysis is to perform an investigation is to explore how departure delays and air time may be associated with arrival delays, focusing specifically on these three carriers. While carrier is included as a categorical variable for potential future use, the primary focus in this assignment will be on building and interpreting models that treat arr\_delay as the response variable. Additional models incorporating carrier or other variables may be explored in subsequent analyses to deepen our understanding of flight delay patterns.

The goal of the 2nd analysis will be discussed in section 3.

# Load necessary packages  
library(nycflights13) ## Data Extraction --- E  
library(dplyr) ## Data Transformation --- T  
library(tidyr) ## Data Transformation --- T  
library(ggplot2) ## Data Visualization --- V  
library(ggfortify) ## Data Visualization --- V  
library(MASS) ## Data Analysis --- A  
library(leaps) ## Data Analysis --- A  
library(lmtest) ## Data Analysis --- A  
library(broom) ## Data Analysis --- A  
library(car) ## Data Analysis --- A

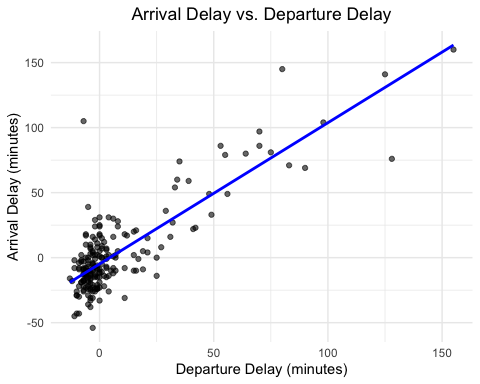
# Set seed for reproducibility  
set.seed(324)   
  
# Sample 200 rows and select relevant variables  
flight\_df<- flights %>%  
 dplyr::select(arr\_delay, dep\_delay, air\_time, carrier) %>%  
 filter(carrier %in% c("AA","UA","MQ")) %>%   
 drop\_na() %>%   
 sample\_n(200)

## 1) Simple Linear Regression

### 1a. Visualization

ggplot(flight\_df, aes(x = dep\_delay, y = arr\_delay)) +  
 geom\_point(alpha = 0.6) +  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +  
 labs(  
 title = "Arrival Delay vs. Departure Delay",  
 x = "Departure Delay (minutes)",  
 y = "Arrival Delay (minutes)"  
 ) +  
 theme\_minimal() +  
 theme(plot.title = element\_text(hjust = 0.5))

`geom\_smooth()` using formula = 'y ~ x'



### 1b. Analysis

# Fit the simple linear regression model  
model\_1 <- lm(arr\_delay ~ dep\_delay, data = flight\_df)  
  
# View the summary of the model  
summary(model\_1)

Call:  
lm(formula = arr\_delay ~ dep\_delay, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-58.177 -12.847 -1.644 10.086 117.442   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -4.83974 1.40871 -3.436 0.00072 \*\*\*  
dep\_delay 1.08607 0.05296 20.507 < 2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.14 on 198 degrees of freedom  
Multiple R-squared: 0.6799, Adjusted R-squared: 0.6783   
F-statistic: 420.5 on 1 and 198 DF, p-value: < 2.2e-16

## 2) Multiple Linear Regression

We are going to investigate the following models:

| Model # | Formula | Description |
| --- | --- | --- |
| **1** | arr\_delay ~ dep\_delay + air\_time | Both dep\_delay and air\_time are **quantitative**. |
| **2** | arr\_delay ~ dep\_delay + carrier | dep\_delay is **quantitative**, carrier is **categorical**. |
| **3** | arr\_delay ~ dep\_delay + air\_time + dep\_delay:air\_time | Interaction between **two quantitative** variables. |
| **4** | arr\_delay ~ dep\_delay + carrier + dep\_delay:carrier | Interaction between **quantitative** and **categorical** predictor. |
| **5** | arr\_delay ~ dep\_delay\_c + air\_time\_c + dep\_delay\_c:air\_time\_c | Mean-centered dep\_delay and air\_time; includes interaction between centered variables. |
| **6** | arr\_delay ~ dep\_delay + I(dep\_delay^2) | Polynomial regression on dep\_delay. |
| **7** | arr\_delay ~ dep\_delay\_gc + carrier + dep\_delay\_gc:carrier | Group-mean centered dep\_delay within each carrier group; includes interaction. |
| **8** | arr\_delay ~ dep\_delay + air\_time\_group + dep\_delay:air\_time\_group | air\_time binned into 4 categories; interaction with dep\_delay. |

#### Model 1: Two Quantitative Predictors

# Model 1: arr\_delay ~ dep\_delay + air\_time  
model1 <- lm(arr\_delay ~ dep\_delay + air\_time, data = flight\_df)  
summary(model1)

Call:  
lm(formula = arr\_delay ~ dep\_delay + air\_time, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-58.183 -12.816 -1.656 10.065 117.448   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -4.7967868 2.7651353 -1.735 0.0844 .   
dep\_delay 1.0860369 0.0531193 20.445 <2e-16 \*\*\*  
air\_time -0.0002486 0.0137578 -0.018 0.9856   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.19 on 197 degrees of freedom  
Multiple R-squared: 0.6799, Adjusted R-squared: 0.6766   
F-statistic: 209.2 on 2 and 197 DF, p-value: < 2.2e-16

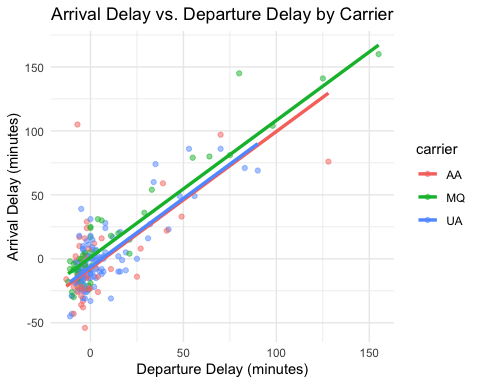
#### Model 2: One Quant + One Categorical

# Model 2: arr\_delay ~ dep\_delay + carrier  
model2 <- lm(arr\_delay ~ dep\_delay + carrier, data = flight\_df)  
summary(model2)

Call:  
lm(formula = arr\_delay ~ dep\_delay + carrier, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-53.515 -11.684 -2.341 8.850 120.059   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -7.56294 2.74404 -2.756 0.0064 \*\*   
dep\_delay 1.07092 0.05269 20.325 <2e-16 \*\*\*  
carrierMQ 8.67699 3.82445 2.269 0.0244 \*   
carrierUA 1.23221 3.31798 0.371 0.7108   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 18.92 on 196 degrees of freedom  
Multiple R-squared: 0.6903, Adjusted R-squared: 0.6855   
F-statistic: 145.6 on 3 and 196 DF, p-value: < 2.2e-16

# For each carrier, get the min and max dep\_delay  
df\_range <- flight\_df %>%  
 group\_by(carrier) %>%  
 summarize(  
 dep\_min = min(dep\_delay, na.rm = TRUE),  
 dep\_max = max(dep\_delay, na.rm = TRUE)  
 ) %>%  
 pivot\_longer(cols = c(dep\_min, dep\_max), names\_to = NULL, values\_to = "dep\_delay")  
  
# Compute the predicted arr\_delay at those endpoints  
df\_range <- df\_range %>%  
 mutate(arr\_pred = predict(model2, newdata = df\_range))  
  
# Plot points and parallel lines truncated to each group's range  
ggplot(flight\_df, aes(x = dep\_delay, y = arr\_delay, color = carrier)) +  
 geom\_point(alpha = 0.5) +  
 geom\_line(  
 data = df\_range,  
 aes(x = dep\_delay, y = arr\_pred, color = carrier),  
 size = 1.2  
 ) +  
 labs(  
 title = "Arrival Delay vs. Departure Delay by Carrier",  
 x = "Departure Delay (minutes)",  
 y = "Arrival Delay (minutes)"  
 ) +  
 theme\_minimal() +  
 theme(plot.title = element\_text(hjust = 0.5))

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
ℹ Please use `linewidth` instead.



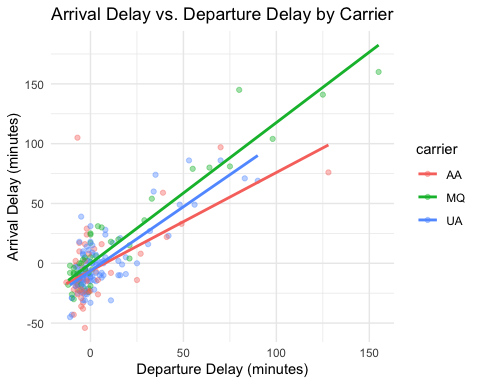
#### Model 3: Quant + Quant + Interaction

# Model 3: arr\_delay ~ dep\_delay \* air\_time  
model3 <- lm(arr\_delay ~ dep\_delay \* air\_time, data = flight\_df)  
summary(model3)

Call:  
lm(formula = arr\_delay ~ dep\_delay \* air\_time, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-58.658 -11.892 -1.652 9.882 116.749   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -5.8710193 2.8600281 -2.053 0.0414 \*   
dep\_delay 1.2075225 0.1006721 11.995 <2e-16 \*\*\*  
air\_time 0.0069649 0.0146336 0.476 0.6346   
dep\_delay:air\_time -0.0008914 0.0006281 -1.419 0.1574   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.14 on 196 degrees of freedom  
Multiple R-squared: 0.6831, Adjusted R-squared: 0.6783   
F-statistic: 140.9 on 3 and 196 DF, p-value: < 2.2e-16

# Plot points and interaction lines truncated to overall dep\_delay range  
ggplot(flight\_df, aes(x = dep\_delay, y = arr\_delay, color = carrier)) +  
 geom\_point(alpha = 0.4) +  
 geom\_smooth(method = "lm", se = FALSE) +  
 labs(  
 title = "Arrival Delay vs. Departure Delay by Carrier",  
 x = "Departure Delay (minutes)",  
 y = "Arrival Delay (minutes)"  
 ) +  
 theme\_minimal() +  
 theme(plot.title = element\_text(hjust = 0.5))

`geom\_smooth()` using formula = 'y ~ x'



#### Model 4: Quant + Categorical + Interaction

# Model 4: arr\_delay ~ dep\_delay \* carrier  
model4 <- lm(arr\_delay ~ dep\_delay \* carrier, data = flight\_df)  
summary(model4)

Call:  
lm(formula = arr\_delay ~ dep\_delay \* carrier, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-45.214 -11.480 -2.206 8.506 117.075   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -6.31937 2.75239 -2.296 0.02275 \*   
dep\_delay 0.82221 0.11095 7.411 3.77e-12 \*\*\*  
carrierMQ 6.12013 3.90036 1.569 0.11825   
carrierUA -0.01569 3.37527 -0.005 0.99630   
dep\_delay:carrierMQ 0.35622 0.13285 2.681 0.00796 \*\*   
dep\_delay:carrierUA 0.24943 0.14896 1.674 0.09565 .   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 18.68 on 194 degrees of freedom  
Multiple R-squared: 0.7014, Adjusted R-squared: 0.6937   
F-statistic: 91.12 on 5 and 194 DF, p-value: < 2.2e-16

#### Model 5: Centered Interaction (both quant)

# Model 5: Center and fit interaction  
flight\_df <- flight\_df %>%  
 mutate(  
 dep\_delay\_c = dep\_delay - mean(dep\_delay, na.rm = TRUE),  
 air\_time\_c = air\_time - mean(air\_time, na.rm = TRUE)  
 )  
  
model5 <- lm(arr\_delay ~ dep\_delay\_c \* air\_time\_c, data = flight\_df)  
summary(model5)

Call:  
lm(formula = arr\_delay ~ dep\_delay\_c \* air\_time\_c, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-58.658 -11.892 -1.652 9.882 116.749   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 3.1016811 1.3542813 2.290 0.0231 \*   
dep\_delay\_c 1.0542672 0.0575181 18.329 <2e-16 \*\*\*  
air\_time\_c 0.0003910 0.0137299 0.028 0.9773   
dep\_delay\_c:air\_time\_c -0.0008914 0.0006281 -1.419 0.1574   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.14 on 196 degrees of freedom  
Multiple R-squared: 0.6831, Adjusted R-squared: 0.6783   
F-statistic: 140.9 on 3 and 196 DF, p-value: < 2.2e-16

#### Model 6: Polynomial Term

# Model 6: arr\_delay ~ dep\_delay + dep\_delay^2  
model6 <- lm(arr\_delay ~ dep\_delay + I(dep\_delay^2), data = flight\_df)  
summary(model6)

Call:  
lm(formula = arr\_delay ~ dep\_delay + I(dep\_delay^2), data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-52.17 -12.59 -1.40 10.36 118.33   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -4.805047 1.407920 -3.413 0.00078 \*\*\*  
dep\_delay 1.208581 0.119318 10.129 < 2e-16 \*\*\*  
I(dep\_delay^2) -0.001326 0.001158 -1.146 0.25334   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.13 on 197 degrees of freedom  
Multiple R-squared: 0.682, Adjusted R-squared: 0.6788   
F-statistic: 211.3 on 2 and 197 DF, p-value: < 2.2e-16

#### Model 7: Group-Centered by Carrier

# Model 7: Group-centered dep\_delay within carrier  
flight\_df <- flight\_df %>%  
 group\_by(carrier) %>%  
 mutate(dep\_delay\_gc = dep\_delay - mean(dep\_delay, na.rm = TRUE)) %>%  
 ungroup()  
  
model7 <- lm(arr\_delay ~ dep\_delay\_gc \* carrier, data = flight\_df)  
summary(model7)

Call:  
lm(formula = arr\_delay ~ dep\_delay\_gc \* carrier, data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-45.214 -11.480 -2.206 8.506 117.075   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -2.2083 2.6959 -0.819 0.41371   
dep\_delay\_gc 0.8222 0.1109 7.411 3.77e-12 \*\*\*  
carrierMQ 16.4044 3.7561 4.367 2.04e-05 \*\*\*  
carrierUA 2.3667 3.2744 0.723 0.47068   
dep\_delay\_gc:carrierMQ 0.3562 0.1328 2.681 0.00796 \*\*   
dep\_delay\_gc:carrierUA 0.2494 0.1490 1.674 0.09565 .   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 18.68 on 194 degrees of freedom  
Multiple R-squared: 0.7014, Adjusted R-squared: 0.6937   
F-statistic: 91.12 on 5 and 194 DF, p-value: < 2.2e-16

#### Model 8: Piecewise Regression (binned air\_time)

# Model 8: Bin air\_time into 4 groups  
flight\_df <- flight\_df %>%  
 mutate(air\_time\_group = ntile(air\_time, 4))  
  
model8 <- lm(arr\_delay ~ dep\_delay \* as.factor(air\_time\_group), data = flight\_df)  
summary(model8)

Call:  
lm(formula = arr\_delay ~ dep\_delay \* as.factor(air\_time\_group),   
 data = flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-44.158 -11.221 -1.920 9.737 115.668   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -3.39524 2.82889 -1.200 0.2315   
dep\_delay 1.25462 0.09007 13.929 <2e-16 \*\*\*  
as.factor(air\_time\_group)2 -4.42297 3.98398 -1.110 0.2683   
as.factor(air\_time\_group)3 -0.04086 3.91649 -0.010 0.9917   
as.factor(air\_time\_group)4 -1.14278 4.06954 -0.281 0.7792   
dep\_delay:as.factor(air\_time\_group)2 -0.25480 0.12357 -2.062 0.0406 \*   
dep\_delay:as.factor(air\_time\_group)3 -0.22153 0.17157 -1.291 0.1982   
dep\_delay:as.factor(air\_time\_group)4 -0.28018 0.17177 -1.631 0.1045   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19 on 192 degrees of freedom  
Multiple R-squared: 0.694, Adjusted R-squared: 0.6828   
F-statistic: 62.21 on 7 and 192 DF, p-value: < 2.2e-16

## 3) Model Selection Techniques & Metrics

This analysis uses the **second random sample of 200 observations** from the nycflights13 dataset, again focusing on flights from three major carriers: American Airlines (AA), United Airlines (UA), and Envoy Air (MQ). This time, we expand our investigation by considering **five explanatory variables**: dep\_delay, air\_time, distance, hour, and day. These variables offer a mix of timing, duration, and scheduling information that may explain variation in the response variable, arr\_delay. By working with a fresh sample, we simulate the real-world variability encountered in practice and apply techniques like **stepwise regression** and **best subset selection** to identify the most parsimonious and predictive subset of variables for modeling arrival delay. This approach allows us to evaluate model quality using criteria such as AIC, adjusted R², and BIC.

flight\_mod\_df <- flights %>%  
 dplyr::select(arr\_delay, dep\_delay, air\_time, distance, hour, day, carrier) %>%  
 filter(carrier %in% c("AA", "UA", "MQ")) %>%  
 drop\_na() %>%  
 sample\_n(200)

### 3a. Stepwise Regression

# Start with a full model  
full\_model <- lm(arr\_delay ~ dep\_delay + air\_time + distance + hour + day, data = flight\_mod\_df)  
  
# Stepwise model using both directions (backward & forward)  
step\_model <- stepAIC(full\_model, direction = "both", trace = TRUE)

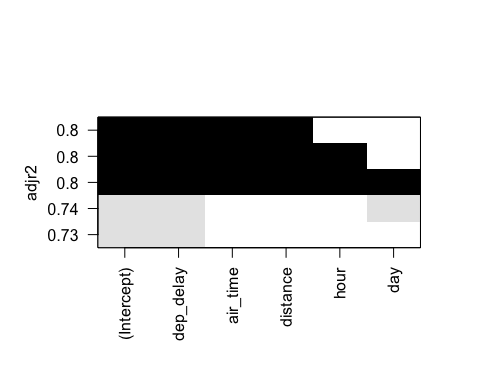
Start: AIC=1204.29  
arr\_delay ~ dep\_delay + air\_time + distance + hour + day  
  
 Df Sum of Sq RSS AIC  
- day 1 118 77751 1202.6  
- hour 1 285 77918 1203.0  
<none> 77633 1204.3  
- air\_time 1 27258 104891 1262.5  
- distance 1 27304 104937 1262.6  
- dep\_delay 1 273600 351233 1504.2  
  
Step: AIC=1202.59  
arr\_delay ~ dep\_delay + air\_time + distance + hour  
  
 Df Sum of Sq RSS AIC  
- hour 1 307 78058 1201.4  
<none> 77751 1202.6  
+ day 1 118 77633 1204.3  
- air\_time 1 27788 105540 1261.7  
- distance 1 27873 105624 1261.9  
- dep\_delay 1 273767 351518 1502.3  
  
Step: AIC=1201.38  
arr\_delay ~ dep\_delay + air\_time + distance  
  
 Df Sum of Sq RSS AIC  
<none> 78058 1201.4  
+ hour 1 307 77751 1202.6  
+ day 1 140 77918 1203.0  
- air\_time 1 27939 105997 1260.6  
- distance 1 28072 106130 1260.8  
- dep\_delay 1 293362 371420 1511.3

# Summary of final selected model  
summary(step\_model)

Call:  
lm(formula = arr\_delay ~ dep\_delay + air\_time + distance, data = flight\_mod\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-31.317 -12.261 -2.722 6.787 116.715   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -20.59690 3.66164 -5.625 6.33e-08 \*\*\*  
dep\_delay 1.04667 0.03856 27.141 < 2e-16 \*\*\*  
air\_time 0.87910 0.10496 8.376 1.04e-14 \*\*\*  
distance -0.11256 0.01341 -8.396 9.18e-15 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.96 on 196 degrees of freedom  
Multiple R-squared: 0.806, Adjusted R-squared: 0.8031   
F-statistic: 271.5 on 3 and 196 DF, p-value: < 2.2e-16

### 3b. Best Subset

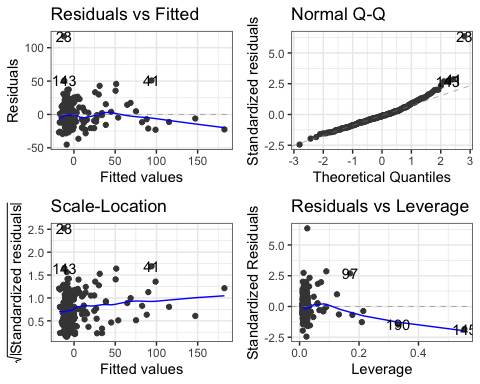
# Fit all subsets model  
subset\_model <- regsubsets(  
 arr\_delay ~ dep\_delay + air\_time + distance + hour + day,  
 data = flight\_mod\_df,  
 nvmax = 5  
)  
  
plot(x = subset\_model, ## Specify Model Fit  
 scale = "adjr2" ## Specify Selection Metric  
)



## 4) Residual Assumptions

In practice, selecting the most appropriate regression model involves balancing model complexity, interpretability, and predictive performance. From the eight models we previously constructed—ranging from simple additive models to interactions and group-centered terms—each offers a different lens on how explanatory variables relate to arrival delay. To illustrate how model assumptions influence model choice, we will now evaluate the residual assumptions of **Model 4**, which includes an interaction between a quantitative predictor (dep\_delay) and a categorical predictor (carrier). This example will highlight the importance of checking conditions like linearity, constant variance, and normality before trusting a model’s inferences.

autoplot(model4) + ## from ggfortify  
 theme\_bw()



resettest(model4) # From lmtest package

RESET test  
  
data: model4  
RESET = 2.5717, df1 = 2, df2 = 192, p-value = 0.07904

dwtest(model4) # From lmtest package

Durbin-Watson test  
  
data: model4  
DW = 1.8679, p-value = 0.1766  
alternative hypothesis: true autocorrelation is greater than 0

shapiro.test(resid(model4)) # Base R

Shapiro-Wilk normality test  
  
data: resid(model4)  
W = 0.91053, p-value = 1.244e-09

bptest(model4) # From lmtest package

studentized Breusch-Pagan test  
  
data: model4  
BP = 7.204, df = 5, p-value = 0.2059

## 5) Transformations

We will focus on simple linear regression using arr\_delay as the response and dep\_delay as the explanatory variable. We will begin by applying an **ad hoc log transformation** to both variables, using log(dep\_delay + 60) and log(arr\_delay + 60) to stabilize variance and address the nonlinear pattern observed in the scatterplot. This transformation is guided by visual diagnostics and practical reasoning rather than a formal algorithm. Following this, we will implement a **Box-Cox transformation**, which systematically searches for the optimal power transformation of the response variable based on model fit. Together, these approaches will allow us to compare how different transformations impact model assumptions and interpretability.

### 5a. Ad Hoc Transformations

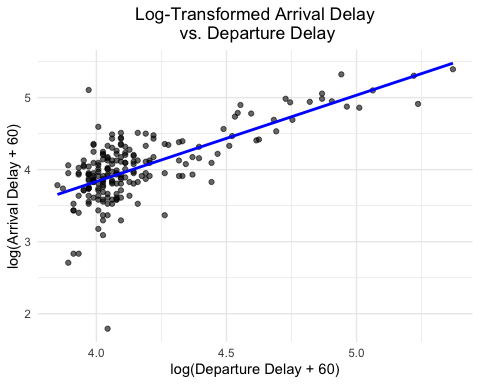
The scatterplot of arr\_delay versus dep\_delay reveals a strong, nonlinear relationship with increasing variance as departure delay increases—a pattern known as *heteroscedasticity*. The spread of arrival delays appears tightly clustered for small values of dep\_delay, but fans out widely for larger delays. This violates the constant variance assumption of linear regression and suggests that our model’s residuals may be non-constant and potentially non-normal. To stabilize the variance and improve linearity, we consider applying transformations to the predictor and/or response variable. In doing so, we add 60 to both arr\_delay and dep\_delay before taking logarithms to avoid issues with log(0) or negative values. Importantly, this shift is a **linear transformation** and does not affect the underlying relationship between the variables—it simply rescales the data to make log transformation valid.

# Add small constant to avoid log(0) issues  
mod\_2\_flight\_df <- flight\_df %>%  
 mutate(  
 log\_arr\_delay = log(arr\_delay + 60),  
 log\_dep\_delay = log(dep\_delay + 60)  
 ) %>%   
 mutate(arr\_delay\_pos = arr\_delay + 60)   
  
# Fit the transformed model  
model\_loglog <- lm(log\_arr\_delay ~ log\_dep\_delay, data = mod\_2\_flight\_df)  
summary(model\_loglog)

Call:  
lm(formula = log\_arr\_delay ~ log\_dep\_delay, data = mod\_2\_flight\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-2.09507 -0.16881 0.02414 0.20912 1.30631   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -0.9582 0.3655 -2.621 0.00944 \*\*   
log\_dep\_delay 1.1984 0.0876 13.680 < 2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 0.3424 on 198 degrees of freedom  
Multiple R-squared: 0.4859, Adjusted R-squared: 0.4833   
F-statistic: 187.1 on 1 and 198 DF, p-value: < 2.2e-16

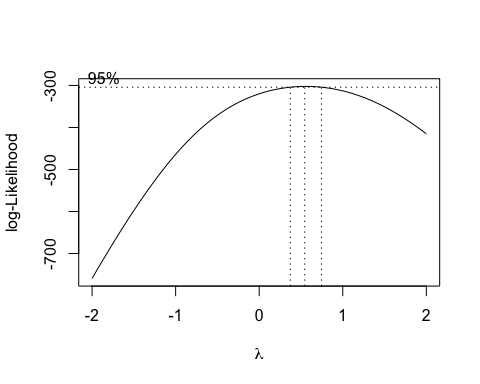
ggplot(mod\_2\_flight\_df, aes(x = log\_dep\_delay, y = log\_arr\_delay)) +  
 geom\_point(alpha = 0.6) +  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +  
 labs(  
 title = "Log-Transformed Arrival Delay\n vs. Departure Delay",  
 x = "log(Departure Delay + 60)",  
 y = "log(Arrival Delay + 60)"  
 ) +  
 theme\_minimal() +  
 theme(plot.title = element\_text(hjust = 0.5))

`geom\_smooth()` using formula = 'y ~ x'



### 5b. Power Transformations – Box-Cox

# Fit the original model again  
model\_boxcox <- lm(arr\_delay\_pos ~ dep\_delay, data = mod\_2\_flight\_df)  
  
# Run Box-Cox to suggest power transformation for Y  
# Run Box-Cox and save result  
bc\_result <- boxcox(model\_boxcox, lambda = seq(-2, 2, 0.1))



bc\_result

$x  
 [1] -2.00000000 -1.95959596 -1.91919192 -1.87878788 -1.83838384 -1.79797980  
 [7] -1.75757576 -1.71717172 -1.67676768 -1.63636364 -1.59595960 -1.55555556  
 [13] -1.51515152 -1.47474747 -1.43434343 -1.39393939 -1.35353535 -1.31313131  
 [19] -1.27272727 -1.23232323 -1.19191919 -1.15151515 -1.11111111 -1.07070707  
 [25] -1.03030303 -0.98989899 -0.94949495 -0.90909091 -0.86868687 -0.82828283  
 [31] -0.78787879 -0.74747475 -0.70707071 -0.66666667 -0.62626263 -0.58585859  
 [37] -0.54545455 -0.50505051 -0.46464646 -0.42424242 -0.38383838 -0.34343434  
 [43] -0.30303030 -0.26262626 -0.22222222 -0.18181818 -0.14141414 -0.10101010  
 [49] -0.06060606 -0.02020202 0.02020202 0.06060606 0.10101010 0.14141414  
 [55] 0.18181818 0.22222222 0.26262626 0.30303030 0.34343434 0.38383838  
 [61] 0.42424242 0.46464646 0.50505051 0.54545455 0.58585859 0.62626263  
 [67] 0.66666667 0.70707071 0.74747475 0.78787879 0.82828283 0.86868687  
 [73] 0.90909091 0.94949495 0.98989899 1.03030303 1.07070707 1.11111111  
 [79] 1.15151515 1.19191919 1.23232323 1.27272727 1.31313131 1.35353535  
 [85] 1.39393939 1.43434343 1.47474747 1.51515152 1.55555556 1.59595960  
 [91] 1.63636364 1.67676768 1.71717172 1.75757576 1.79797980 1.83838384  
 [97] 1.87878788 1.91919192 1.95959596 2.00000000  
  
$y  
 [1] -759.1496 -745.3815 -731.7192 -718.1683 -704.7345 -691.4232 -678.2402  
 [8] -665.1917 -652.2842 -639.5245 -626.9195 -614.4766 -602.2031 -590.1068  
 [15] -578.1955 -566.4773 -554.9601 -543.6522 -532.5617 -521.6966 -511.0648  
 [22] -500.6742 -490.5320 -480.6454 -471.0211 -461.6652 -452.5833 -443.7802  
 [29] -435.2603 -427.0270 -419.0829 -411.4301 -404.0694 -397.0013 -390.2249  
 [36] -383.7389 -377.5414 -371.6293 -365.9994 -360.6474 -355.5690 -350.7592  
 [43] -346.2128 -341.9242 -337.8878 -334.0979 -330.5485 -327.2341 -324.1488  
 [50] -321.2874 -318.6443 -316.2147 -313.9937 -311.9769 -310.1603 -308.5399  
 [57] -307.1123 -305.8746 -304.8239 -303.9579 -303.2746 -302.7724 -302.4498  
 [64] -302.3058 -302.3397 -302.5508 -302.9390 -303.5043 -304.2466 -305.1663  
 [71] -306.2639 -307.5397 -308.9944 -310.6286 -312.4428 -314.4377 -316.6136  
 [78] -318.9711 -321.5104 -324.2317 -327.1348 -330.2197 -333.4858 -336.9325  
 [85] -340.5588 -344.3637 -348.3457 -352.5030 -356.8339 -361.3360 -366.0069  
 [92] -370.8439 -375.8442 -381.0045 -386.3217 -391.7921 -397.4122 -403.1780  
 [99] -409.0858 -415.1318

# Extract the lambda value that maximizes the log-likelihood  
best\_lambda <- bc\_result$x[which.max(bc\_result$y)]  
best\_lambda

[1] 0.5454545

## 6) Miscellaneous Topics

### 6a. Outliers, Leverage, and Influence

best\_model <- lm(arr\_delay ~ dep\_delay + air\_time + distance, data = flight\_mod\_df)  
  
model\_aug <- augment(best\_model)

# Outliers: Standardized residuals > 2 or < -2  
 model\_aug %>%   
 filter(abs(.std.resid) > 2) %>%   
 nrow()

[1] 11

# Leverage: Points with .hat > 2 \* average hat value  
leverage\_cutoff <- 2 \* mean(model\_aug$.hat)  
model\_aug %>%   
 filter(.hat > leverage\_cutoff) %>%   
 nrow()

[1] 14

# Influence: Cook's Distance > 4 / n  
influence\_cutoff <- 4 / nrow(model\_aug)  
model\_aug %>%   
 filter(.cooksd > influence\_cutoff) %>%   
 nrow()

[1] 13

### 6b. Multicollinearity

car::vif(best\_model)

dep\_delay air\_time distance   
 1.032278 48.242936 48.149460

## 7) Statistical Tests

In this section, we use our selected multiple regression model—predicting arr\_delay from dep\_delay, air\_time, and distance—to perform formal statistical inference. We begin with the **omnibus F-test**, which evaluates whether the model as a whole explains a significant amount of variability in arrival delays compared to a model with no predictors. To assess the unique contribution of distance, we conduct a **partial F-test**, comparing the full model to a reduced model without distance. This helps determine whether distance adds meaningful explanatory power beyond dep\_delay and air\_time. Finally, we interpret the results of **individual t-tests** for each coefficient, evaluating whether each predictor is significantly associated with arrival delay while holding the others constant. To visualize these effects, we use **partial regression plots**, which isolate the adjusted relationship between each predictor and the response after accounting for other variables in the model.

# Fit the full model (best model)  
model\_full <- lm(arr\_delay ~ dep\_delay + air\_time + distance, data = flight\_mod\_df)  
  
# Fit the reduced model without 'distance' for partial F-test  
model\_reduced <- lm(arr\_delay ~ dep\_delay + air\_time, data = flight\_mod\_df)

### 7a. Omnibus Test

*“Do these predictors—taken together—explain any variation in arrival delay at all?”*

We use the **global F-test** to test whether at least one of the predictors (dep\_delay, air\_time, distance) is statistically significant, i.e., whether the full model explains more variance than a null model (with just the intercept).

summary(model\_full)

Call:  
lm(formula = arr\_delay ~ dep\_delay + air\_time + distance, data = flight\_mod\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-31.317 -12.261 -2.722 6.787 116.715   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -20.59690 3.66164 -5.625 6.33e-08 \*\*\*  
dep\_delay 1.04667 0.03856 27.141 < 2e-16 \*\*\*  
air\_time 0.87910 0.10496 8.376 1.04e-14 \*\*\*  
distance -0.11256 0.01341 -8.396 9.18e-15 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.96 on 196 degrees of freedom  
Multiple R-squared: 0.806, Adjusted R-squared: 0.8031   
F-statistic: 271.5 on 3 and 196 DF, p-value: < 2.2e-16

### 7b. Partial F-Test

*“Is distance still useful in the model once dep\_delay and air\_time are already included?”* We compare the **full model** (dep\_delay + air\_time + distance) to a **reduced model** (dep\_delay + air\_time) using a partial F-test. This tells us whether the added predictor provides significant additional explanatory power.

# Partial F-test using anova()  
anova(model\_reduced, model\_full)

Analysis of Variance Table  
  
Model 1: arr\_delay ~ dep\_delay + air\_time  
Model 2: arr\_delay ~ dep\_delay + air\_time + distance  
 Res.Df RSS Df Sum of Sq F Pr(>F)   
1 197 106130   
2 196 78058 1 28072 70.487 9.183e-15 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### 7c. Individual t-tests

*“Is each predictor—dep\_delay, air\_time, and distance—significantly associated with arr\_delay when holding the others constant?”*

Each coefficient in the model is tested using a **t-test** to assess whether its effect is different from zero in the context of the multiple regression.

summary(model\_full)

Call:  
lm(formula = arr\_delay ~ dep\_delay + air\_time + distance, data = flight\_mod\_df)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-31.317 -12.261 -2.722 6.787 116.715   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) -20.59690 3.66164 -5.625 6.33e-08 \*\*\*  
dep\_delay 1.04667 0.03856 27.141 < 2e-16 \*\*\*  
air\_time 0.87910 0.10496 8.376 1.04e-14 \*\*\*  
distance -0.11256 0.01341 -8.396 9.18e-15 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.96 on 196 degrees of freedom  
Multiple R-squared: 0.806, Adjusted R-squared: 0.8031   
F-statistic: 271.5 on 3 and 196 DF, p-value: < 2.2e-16

# Create partial regression plots  
avPlots(model\_full,  
 terms = ~ dep\_delay + air\_time + distance,  
 main = "Partial Regression Plots")

