# Major assignment 1: Your submission

This is your assignment template for [AnalyticsX Major assignment 1](https://courses.edx.org/courses/course-v1:AdelaideX+AnalyticsX+1T2021/courseware/e0b375053b6441a08d658133752b3531/b06e32d08342432b8ccd0f2c738f5bb8/1?activate_block_id=block-v1%3AAdelaideX%2BAnalyticsX%2B1T2021%2Btype%40vertical%2Bblock%40457a6f9c5f7444c286c6f3a3e2851bc9). Save this document on our local machine and include all of your work within the relevant sections. Once you’ve completed all five parts of the assignment, upload the document via the submission area on the “[Submit your assignment](https://courses.edx.org/courses/course-v1:AdelaideX+AnalyticsX+1T2021/courseware/e0b375053b6441a08d658133752b3531/b06e32d08342432b8ccd0f2c738f5bb8/7?activate_block_id=block-v1%3AAdelaideX%2BAnalyticsX%2B1T2021%2Btype%40vertical%2Bblock%404de5ad6a293e4d6092d1d5756facfc9d)” page at the end of Major assignment 1.

# Checklist

* Have you shown all of your working?
* Have you given all numbers to the correct number of decimal places?
* Have you included all R output and plots to support your answers where necessary?
* Have you included all of your R code – input and output?
* Have you made sure that all plots and tables each have a meaningful caption?

**Quick links:**

[Major assignment 1: Part 1](#_Major_assignment_1:)

[Major assignment 1: Part 2](#_Major_assignment_1:_1)

[Major assignment 1: Part 3](#_Major_assignment_1:_2)

[Major assignment 1: Part 4](#_Major_assignment_1:_3)

[Major assignment 1: Part 5](#_Major_assignment_1:_4)

# Major assignment 1: Part 1

1. Load in the flights dataset from the nycflights13 package [1 point]

*Your input* code and output code for the flights dataset *from the nycflights13 package go here:*

**Import Libraries**

library(tidyverse)

library(modelr)

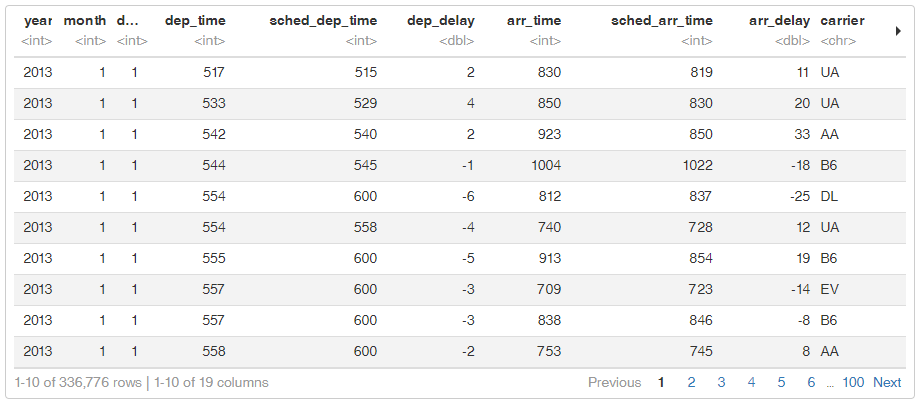
library(nycflights13)

**Loads the flights dataset into a local variable**

flights <- nycflights13::flights

flights

Table 1. Loaded flights dataset.

****

1. Produce a scatterplot of dep\_delay against arr\_delay [3 points]

Your plot, code and caption go here:

ggplot(flights, aes(x=dep\_delay, y=arr\_delay)) +

  labs(title="Flight Delays", x="Departure Delay", y="Arrival Delay") +

  geom\_point() +

  geom\_smooth()

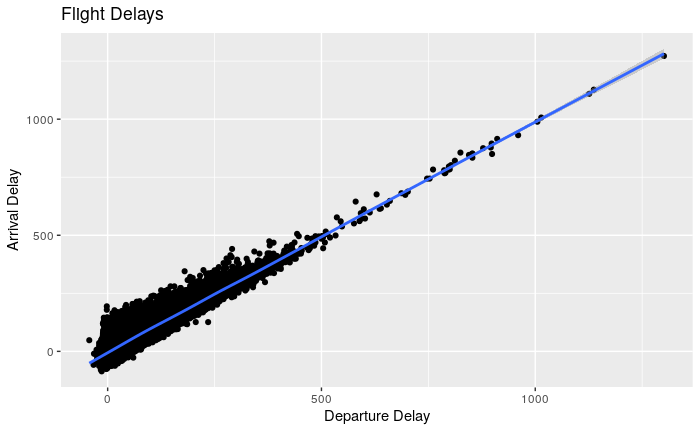


Figure 1. Departure Delay vs Arrival Delay.

1. Fit a linear regression model with arr\_delay as the response variable and dep\_delay as the predictor, and obtain the intercept and slope of the fitted model. [3 points]

Your code goes here:

flights.lm <- lm(arr\_delay ~ dep\_delay, data=flights)

summary(flights.lm)

Table 2. Linear Regression of Flight Delays

|  |
| --- |
| Call:  lm(formula = arr\_delay ~ dep\_delay, data = flights)  Residuals:  Min 1Q Median 3Q Max  -107.587 -11.005 -1.883 8.938 201.938  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -5.8994935 0.0330195 -178.7 <2e-16 \*\*\*  dep\_delay 1.0190929 0.0007864 1295.8 <2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 18.03 on 327344 degrees of freedom  (9430 observations deleted due to missingness)  Multiple R-squared: 0.8369, Adjusted R-squared: 0.8369  F-statistic: 1.679e+06 on 1 and 327344 DF, p-value: < 2.2e-16 |

**Intercept:** -5.8995  
**Slope:** 1.0191

1. Produce a residual versus fitted plot for the fitted model [1 point]

Your code and plot go here:

plot(flights.lm, which = 1)

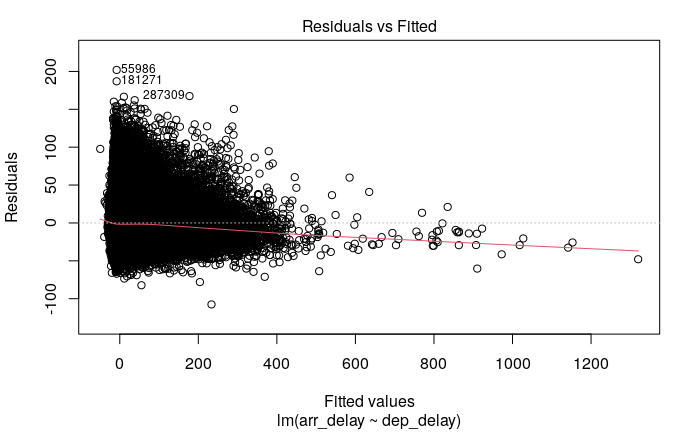


Figure 2. Residuals versus Fitted Plot

1. Produce a normal QQ\_plot for the fitted model [1 point]

Your code and plot go here:

plot(flights.lm, which=2)

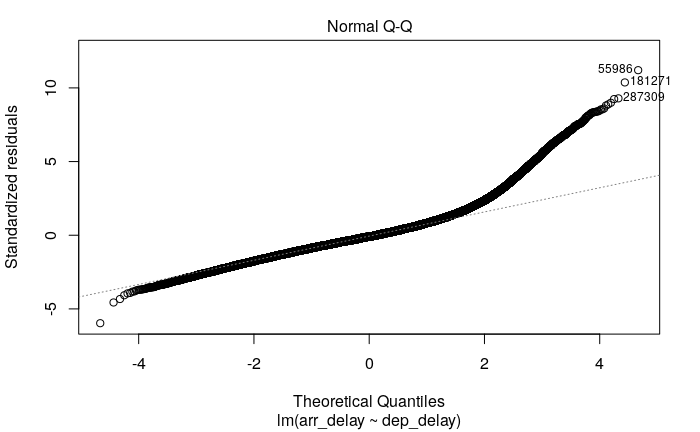


Figure 3. Normal Q-Q Plot.

6. Predict the expected arrival delay for flights with a departure delay of 100 minutes [1 point]

Your code (as evidence of your answer to Part 1, Question 6) goes here:

newdata <- data.frame(dep\_delay=c(100))

predict(flights.lm, newdata=newdata)

Table 3. Predicated Arrival Delay

|  |
| --- |
| 1  96.0098 |

# Major assignment 1: Part 2

1. Load in the flights dataset from the nycflights13 package [1 point]

Your input code and output code for the flights dataset from the nycflights13 package go here:

**Import Libraries**

library(tidyverse)

library(modelr)

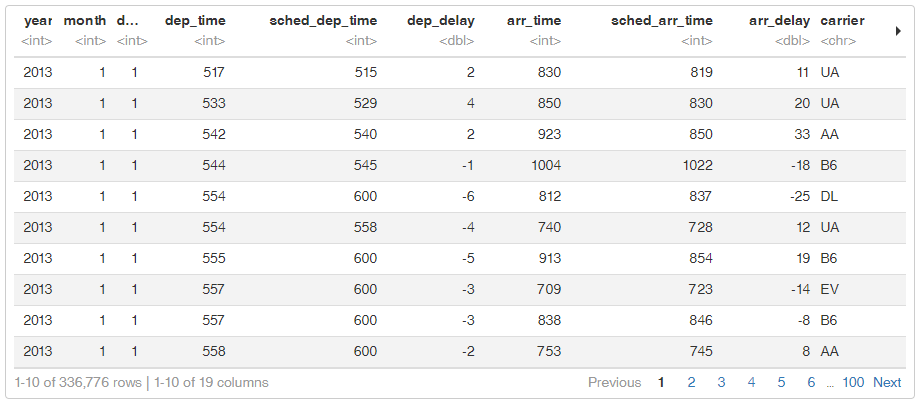
library(nycflights13)

**Loads the flights dataset into a local variable**

flights <- nycflights13::flights

flights

Table 4. Flights Dataset - First 10 Rows



1. Produce a scatterplot of arr\_delay against dep\_delay with colours for the different origins. Add a linear regression line to the plot for each origin. [2 points]

Your plot and code go here:

ggplot(flights,mapping = aes(x=dep\_delay, y=arr\_delay)) +

  labs(title="Flight Delays", x="Departure Delay", y="Arrival Delay") +

  geom\_point(aes(color = origin)) +

  geom\_smooth(method = 'lm', mapping = aes(color=origin))

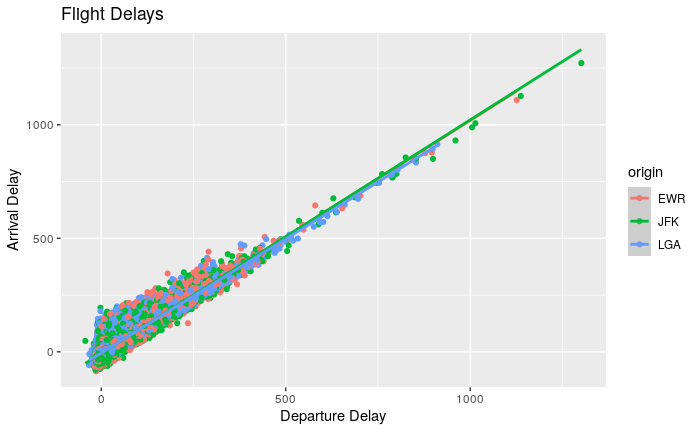


Figure 4 Arrival Delays vs Departure Delays.

1. Fit a linear model with arr\_delay as the response variable; dep\_delay as a predictor; origin as another predictor; and an interaction term between dep\_delay and origin. [2 points]

Your input and output code goes here:

flights.lm2 <- lm(arr\_delay ~ dep\_delay + origin + dep\_delay:origin, data=flights)

summary(flights.lm2)

Table 5. Model Summary.

|  |
| --- |
| Call:  lm(formula = arr\_delay ~ dep\_delay + origin + dep\_delay:origin,  data = flights)  Residuals:  Min 1Q Median 3Q Max  -108.706 -10.970 -1.588 8.800 202.872  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) -6.284653 0.055989 -112.249 < 2e-16 \*\*\*  dep\_delay 1.025491 0.001277 802.880 < 2e-16 \*\*\*  originJFK -0.530664 0.079944 -6.638 3.19e-11 \*\*\*  originLGA 1.745625 0.080946 21.565 < 2e-16 \*\*\*  dep\_delay:originJFK 0.003052 0.001898 1.608 0.108  dep\_delay:originLGA -0.021998 0.001909 -11.525 < 2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 18 on 327340 degrees of freedom  (9430 observations deleted due to missingness)  Multiple R-squared: 0.8373, Adjusted R-squared: 0.8373  F-statistic: 3.369e+05 on 5 and 327340 DF, p-value: < 2.2e-16 |

1. Produce a table of the coefficients for each term in the model. [1 point]

Your input and output code go here:

summary(flights.lm2)$coefficients

Table 6. Coefficients Table.

|  |
| --- |
| Estimate Std. Error t value Pr(>|t|)  (Intercept) -6.284653169 0.055988538 -112.248925 0.000000e+00  dep\_delay 1.025491061 0.001277265 802.880146 0.000000e+00  originJFK -0.530663701 0.079943937 -6.637948 3.185730e-11  originLGA 1.745625136 0.080945687 21.565388 4.481705e-103  dep\_delay:originJFK 0.003052052 0.001898006 1.608030 1.078295e-01  dep\_delay:originLGA -0.021997840 0.001908722 -11.524903 1.001837e-30 |

# Major assignment 1: Part 3

1. Load in the flights dataset from the nycflights13 package [1 point]

Your input code and output code for the flights dataset from the nycflights13 package go here:

**Import Libraries**

library(tidyverse)

library(modelr)

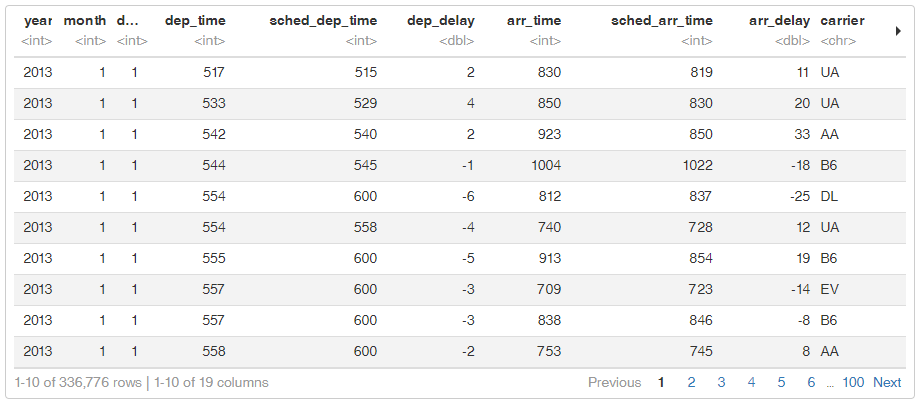
library(nycflights13)

**Loads the flights dataset into a local variable**

flights <- nycflights13::flights

flights

Table 7. Loaded Flights Dataset



1. Produce a nested dataframe with a row for each carrier [2 points]

Your input and output code goes here:

flights.nested <- flights %>%

  group\_by(carrier) %>%

  nest()

flights.nested

Table 8. Flights nested by carrier.

|  |
| --- |
| # A tibble: 16 x 2  # Groups: carrier [16]  carrier data  <chr> <list>  1 UA <tibble [58,665 × 18]>  2 AA <tibble [32,729 × 18]>  3 B6 <tibble [54,635 × 18]>  4 DL <tibble [48,110 × 18]>  5 EV <tibble [54,173 × 18]>  6 MQ <tibble [26,397 × 18]>  7 US <tibble [20,536 × 18]>  8 WN <tibble [12,275 × 18]>  9 VX <tibble [5,162 × 18]>  10 FL <tibble [3,260 × 18]>  11 AS <tibble [714 × 18]>  12 9E <tibble [18,460 × 18]>  13 F9 <tibble [685 × 18]>  14 HA <tibble [342 × 18]>  15 YV <tibble [601 × 18]>  16 OO <tibble [32 × 18]> |

1. Fit a linear model, for each carrier, of arrival delay regressed on departure delay. [5 points]

Your input and output code goes here:

flights\_model <- function(df) {

  lm(arr\_delay ~ dep\_delay, data=df)

}

flights.model  <- flights.nested %>%

  mutate(model = map(data, flights\_model),

         coef = map(model, broom::tidy))

flights.model

Table 9. Models for each carrier.

|  |
| --- |
| # A tibble: 16 x 4  # Groups: carrier [16]  carrier data model coef  <chr> <list> <list> <list>  1 UA <tibble [58,665 × 18]> <lm> <tibble [2 × 5]>  2 AA <tibble [32,729 × 18]> <lm> <tibble [2 × 5]>  3 B6 <tibble [54,635 × 18]> <lm> <tibble [2 × 5]>  4 DL <tibble [48,110 × 18]> <lm> <tibble [2 × 5]>  5 EV <tibble [54,173 × 18]> <lm> <tibble [2 × 5]>  6 MQ <tibble [26,397 × 18]> <lm> <tibble [2 × 5]>  7 US <tibble [20,536 × 18]> <lm> <tibble [2 × 5]>  8 WN <tibble [12,275 × 18]> <lm> <tibble [2 × 5]>  9 VX <tibble [5,162 × 18]> <lm> <tibble [2 × 5]>  10 FL <tibble [3,260 × 18]> <lm> <tibble [2 × 5]>  11 AS <tibble [714 × 18]> <lm> <tibble [2 × 5]>  12 9E <tibble [18,460 × 18]> <lm> <tibble [2 × 5]>  13 F9 <tibble [685 × 18]> <lm> <tibble [2 × 5]>  14 HA <tibble [342 × 18]> <lm> <tibble [2 × 5]>  15 YV <tibble [601 × 18]> <lm> <tibble [2 × 5]>  16 OO <tibble [32 × 18]> <lm> <tibble [2 × 5]> |

yv <- flights.model %>%

  filter(carrier == "YV")

yv$model[[1]]

Table 10. Carrier YV Model

|  |
| --- |
| Call:  lm(formula = arr\_delay ~ dep\_delay, data = df)  Coefficients:  (Intercept) dep\_delay  -3.707 1.019 |

1. Produce a table of the coefficients for each carrier model, and display the portion of the table for carrier YV. [3 points]

Your input and output code goes here:

flights.coef  <- flights.model %>% unnest(coef)

flights.coef %>%

  filter(carrier == "YV")

Table 11. Carrier YV Coefficients.

|  |
| --- |
| # A tibble: 2 x 8  # Groups: carrier [1]  carrier data model term estimate std.error statistic p.value  <chr> <list> <list> <chr> <dbl> <dbl> <dbl> <dbl>  1 YV <tibble [601 × 18]> <lm> (Intercept) -3.71 0.782 -4.74 2.73e- 6  2 YV <tibble [601 × 18]> <lm> dep\_delay 1.02 0.0149 68.6 2.27e-269 |

flights.coef %>% print(n = Inf)

Table 12. All Coefficients.

|  |
| --- |
| # A tibble: 32 x 8  # Groups: carrier [16]  carrier data model term estimate std.error statistic p.value  <chr> <list> <list> <chr> <dbl> <dbl> <dbl> <dbl>  1 UA <tibble [58,665 × 18]> <lm> (Intercept) -8.71 0.0837 -104. 0  2 UA <tibble [58,665 × 18]> <lm> dep\_delay 1.02 0.00223 458. 0  3 AA <tibble [32,729 × 18]> <lm> (Intercept) -8.33 0.110 -75.4 0  4 AA <tibble [32,729 × 18]> <lm> dep\_delay 1.01 0.00288 352. 0  5 B6 <tibble [54,635 × 18]> <lm> (Intercept) -3.78 0.0785 -48.2 0  6 B6 <tibble [54,635 × 18]> <lm> dep\_delay 1.02 0.00194 527. 0  7 DL <tibble [48,110 × 18]> <lm> (Intercept) -7.70 0.0888 -86.8 0  8 DL <tibble [48,110 × 18]> <lm> dep\_delay 1.01 0.00218 465. 0  9 EV <tibble [54,173 × 18]> <lm> (Intercept) -4.50 0.0727 -61.8 0  10 EV <tibble [54,173 × 18]> <lm> dep\_delay 1.02 0.00144 710. 0  11 MQ <tibble [26,397 × 18]> <lm> (Intercept) 0.132 0.110 1.20 2.31e- 1  12 MQ <tibble [26,397 × 18]> <lm> dep\_delay 1.02 0.00272 374. 0  13 US <tibble [20,536 × 18]> <lm> (Intercept) -1.74 0.116 -15.0 1.25e- 50  14 US <tibble [20,536 × 18]> <lm> dep\_delay 1.03 0.00411 251. 0  15 WN <tibble [12,275 × 18]> <lm> (Intercept) -8.22 0.166 -49.6 0  16 WN <tibble [12,275 × 18]> <lm> dep\_delay 1.01 0.00355 285. 0  17 VX <tibble [5,162 × 18]> <lm> (Intercept) -11.4 0.299 -38.2 2.39e-281  18 VX <tibble [5,162 × 18]> <lm> dep\_delay 1.03 0.00653 158. 0  19 FL <tibble [3,260 × 18]> <lm> (Intercept) 1.78 0.298 5.98 2.44e- 9  20 FL <tibble [3,260 × 18]> <lm> dep\_delay 0.985 0.00535 184. 0  21 AS <tibble [714 × 18]> <lm> (Intercept) -15.6 0.762 -20.5 1.89e- 73  22 AS <tibble [714 × 18]> <lm> dep\_delay 0.972 0.0239 40.7 1.26e-187  23 9E <tibble [18,460 × 18]> <lm> (Intercept) -9.43 0.150 -62.7 0  24 9E <tibble [18,460 × 18]> <lm> dep\_delay 1.02 0.00311 329. 0  25 F9 <tibble [685 × 18]> <lm> (Intercept) 2.06 0.912 2.26 2.39e- 2  26 F9 <tibble [685 × 18]> <lm> dep\_delay 0.983 0.0148 66.6 6.79e-300  27 HA <tibble [342 × 18]> <lm> (Intercept) -11.6 1.25 -9.31 1.67e- 18  28 HA <tibble [342 × 18]> <lm> dep\_delay 0.965 0.0169 57.2 1.59e-176  29 YV <tibble [601 × 18]> <lm> (Intercept) -3.71 0.782 -4.74 2.73e- 6  30 YV <tibble [601 × 18]> <lm> dep\_delay 1.02 0.0149 68.6 2.27e-269  31 OO <tibble [32 × 18]> <lm> (Intercept) -1.73 2.62 -0.659 5.15e- 1  32 OO <tibble [32 × 18]> <lm> dep\_delay 1.09 0.0594 18.3 9.79e- 17 |

# Major assignment 1: Part 4

1. Load in the flights dataset from the nycflights13 package [1 point]

Your input code and output code for the flights dataset from the nycflights13 package go here:

**Import Libraries**

library(tidyverse)

library(modelr)

library(nycflights13)

**Loads the flights dataset into a local variable**

flights <- nycflights13::flights

flights

Table 13. Flights Tibble

|  |
| --- |
| # A tibble: 336,776 x 19  year month day dep\_time sched\_dep\_time dep\_delay arr\_time sched\_arr\_time arr\_delay carrier flight tailnum origin dest  <int> <int> <int> <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr> <chr> <chr>  1 2013 1 1 517 515 2 830 819 11 UA 1545 N14228 EWR IAH  2 2013 1 1 533 529 4 850 830 20 UA 1714 N24211 LGA IAH  3 2013 1 1 542 540 2 923 850 33 AA 1141 N619AA JFK MIA  4 2013 1 1 544 545 -1 1004 1022 -18 B6 725 N804JB JFK BQN  5 2013 1 1 554 600 -6 812 837 -25 DL 461 N668DN LGA ATL  6 2013 1 1 554 558 -4 740 728 12 UA 1696 N39463 EWR ORD  7 2013 1 1 555 600 -5 913 854 19 B6 507 N516JB EWR FLL  8 2013 1 1 557 600 -3 709 723 -14 EV 5708 N829AS LGA IAD  9 2013 1 1 557 600 -3 838 846 -8 B6 79 N593JB JFK MCO  10 2013 1 1 558 600 -2 753 745 8 AA 301 N3ALAA LGA ORD  # … with 336,766 more rows, and 5 more variables: air\_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time\_hour <dttm> |

1. Create a new variable called delayed that has the value 1 if the arrival delay is greater than zero, and value 0 if the arrival delay is less than or equal to zero. Count the number of delayed flights. [4 points]

Your input and output code goes here:

**Add Delayed Variable**

flights$delayed <- as.numeric(flights$arr\_delay > 0)

flights

**Overview of the Delayed Status**

flights %>%

  group\_by(delayed) %>%

  summarise(count=n())

Table 14. Flight Delayed Indicator

|  |
| --- |
| # A tibble: 3 x 2  delayed count  <dbl> <int>  1 0 194342  2 1 133004  3 NA 9430 |

**Get the total of delayed flights as a variable**

delayed\_flights <- flights %>%

  filter(delayed == 1) %>%

  summarise(total=sum(delayed))

delayed\_total <- delayed\_flights[[1]]

delayed\_total

Table 15. Total number of delayed flights.

|  |
| --- |
| [1] 133004 |

1. Fit a logistic regression model with delayed as the response variable and origin as the predictor. [3 points]

Your input and output code goes here:

flights.glm <- glm(delayed ~ origin, family=binomial(), flights)

summary(flights.glm)

Table 16. Logistic regression model summary.

|  |
| --- |
| Call:  glm(formula = delayed ~ origin, family = binomial(), data = flights)  Deviance Residuals:  Min 1Q Median 3Q Max  -1.0566 -1.0037 -0.9995 1.3617 1.3664  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) -0.291109 0.005906 -49.29 <2e-16 \*\*\*  originJFK -0.142958 0.008562 -16.70 <2e-16 \*\*\*  originLGA -0.132350 0.008731 -15.16 <2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 442236 on 327345 degrees of freedom  Residual deviance: 441889 on 327343 degrees of freedom  (9430 observations deleted due to missingness)  AIC: 441895  Number of Fisher Scoring iterations: 4 |

1. Produce a table of the coefficients for the logistic regression. [5 points]

Your input and output code goes here:

flights.glm\_coef <- broom::tidy(flights.glm)

flights.glm\_coef

Table 17. Logistic regression coefficients.

|  |
| --- |
| # A tibble: 3 x 5  term estimate std.error statistic p.value  <chr> <dbl> <dbl> <dbl> <dbl>  1 (Intercept) -0.291 0.00591 -49.3 0  2 originJFK -0.143 0.00856 -16.7 1.37e-62  3 originLGA -0.132 0.00873 -15.2 6.62e-52 |

**Delay Probabilities**

newdata <- data.frame(origin=c("LGA", "JFK", "EWR"))

predict(flights.glm, newdata=newdata, type="response")

Table 18. Probability of delay for airports in the same order as the input code.

|  |
| --- |
| 1 2 3  0.3956891 0.3931554 0.4277323 |

# Major assignment 1: Part 5

1. Set the random seed to be 19, then load the flights dataset from the nycflights13 package. Count the number of flights in the dataset that arrived at their destination early. [3 points]

Your input and output code goes here:

library(tidyverse)

library(modelr)

library(nycflights13)

set.seed(19)

flights <- nycflights13::flights

flights$early <- as.numeric(flights$arr\_delay < 0)

early\_flights <- flights %>%

  filter(early == 1) %>%

  summarise(total=sum(early))

early\_total <- early\_flights[[1]]

early\_total

Table 19. Total number of flights arriving early.

|  |
| --- |
| [1] 188933 |

1. Build a model to predict arrival delay using origin, departure delay, flight time, carrier, distance travelled, year, month, day and hour. This is a "full" model compared to those in Parts 1-4 of this assignment. [3 points]

Your input and output code goes here:

flights.model <- lm(arr\_delay ~ origin + dep\_delay + air\_time + carrier + distance + year + month + day + hour, flights)

summary(flights.model)

Table 20. Full model summary.

|  |
| --- |
| Call:  lm(formula = arr\_delay ~ origin + dep\_delay + air\_time + carrier +  distance + year + month + day + hour, data = flights)  Residuals:  Min 1Q Median 3Q Max  -111.890 -9.446 -1.941 6.841 204.901  Coefficients: (1 not defined because of singularities)  Estimate Std. Error t value Pr(>|t|)  (Intercept) -2.263e+01 1.825e-01 -124.001 < 2e-16 \*\*\*  originJFK -8.344e-01 9.034e-02 -9.237 < 2e-16 \*\*\*  originLGA -5.714e-01 8.229e-02 -6.944 3.82e-12 \*\*\*  dep\_delay 1.023e+00 6.844e-04 1494.955 < 2e-16 \*\*\*  air\_time 7.108e-01 2.114e-03 336.190 < 2e-16 \*\*\*  carrierAA 2.464e+00 1.545e-01 15.952 < 2e-16 \*\*\*  carrierAS -6.347e+00 5.993e-01 -10.590 < 2e-16 \*\*\*  carrierB6 9.012e+00 1.364e-01 66.078 < 2e-16 \*\*\*  carrierDL 4.259e+00 1.449e-01 29.392 < 2e-16 \*\*\*  carrierEV 4.905e+00 1.510e-01 32.472 < 2e-16 \*\*\*  carrierF9 8.366e+00 6.048e-01 13.833 < 2e-16 \*\*\*  carrierFL 1.251e+01 3.032e-01 41.264 < 2e-16 \*\*\*  carrierHA 2.218e+01 8.625e-01 25.711 < 2e-16 \*\*\*  carrierMQ 9.924e+00 1.570e-01 63.213 < 2e-16 \*\*\*  carrierOO 8.389e+00 2.842e+00 2.952 0.00316 \*\*  carrierUA 2.465e+00 1.584e-01 15.559 < 2e-16 \*\*\*  carrierUS 8.908e+00 1.669e-01 53.374 < 2e-16 \*\*\*  carrierVX -4.872e-01 2.615e-01 -1.863 0.06247 .  carrierWN -3.783e-01 1.943e-01 -1.947 0.05150 .  carrierYV 6.459e+00 6.689e-01 9.656 < 2e-16 \*\*\*  distance -9.117e-02 2.718e-04 -335.412 < 2e-16 \*\*\*  year NA NA NA NA  month 2.137e-01 7.859e-03 27.198 < 2e-16 \*\*\*  day 2.417e-03 3.044e-03 0.794 0.42716  hour -6.230e-02 5.895e-03 -10.568 < 2e-16 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 15.29 on 327322 degrees of freedom  (9430 observations deleted due to missingness)  Multiple R-squared: 0.8827, Adjusted R-squared: 0.8827  F-statistic: 1.071e+05 on 23 and 327322 DF, p-value: < 2.2e-16 |

1. Use backward elimination starting from this model to create a better linear regression model to predict arrival delay. [3 points]

Your input and output code goes here:

lm.full <- flights.model

lm.null <- lm(arr\_delay ~ 1, data=flights)

step(lm.full, scope=list(lower=lm.null), direction="backward")

Table 21. Backward Elimination Results.

|  |
| --- |
| Start: AIC=1785366  arr\_delay ~ origin + dep\_delay + air\_time + carrier + distance +  year + month + day + hour  Step: AIC=1785366  arr\_delay ~ origin + dep\_delay + air\_time + carrier + distance +  month + day + hour  Df Sum of Sq RSS AIC  - day 1 147 76491645 1785365  <none> 76491498 1785366  - origin 2 20935 76512433 1785452  - hour 1 26098 76517596 1785476  - month 1 172865 76664363 1786103  - carrier 15 3257077 79748574 1798986  - distance 1 26290229 102781726 1882071  - air\_time 1 26412434 102903931 1882460  - dep\_delay 1 522269036 598760534 2458938  Step: AIC=1785365  arr\_delay ~ origin + dep\_delay + air\_time + carrier + distance +  month + hour  Df Sum of Sq RSS AIC  <none> 76491645 1785365  - origin 2 20935 76512580 1785450  - hour 1 26099 76517744 1785474  - month 1 172914 76664559 1786102  - carrier 15 3257137 79748782 1798985  - distance 1 26290595 102782240 1882071  - air\_time 1 26412729 102904374 1882460  - dep\_delay 1 522269717 598761362 2458936  Call:  lm(formula = arr\_delay ~ origin + dep\_delay + air\_time + carrier +  distance + month + hour, data = flights)  Coefficients:  (Intercept) originJFK originLGA dep\_delay air\_time carrierAA carrierAS carrierB6  -22.59259 -0.83443 -0.57141 1.02313 0.71074 2.46440 -6.34648 9.01254  carrierDL carrierEV carrierF9 carrierFL carrierHA carrierMQ carrierOO carrierUA  4.25894 4.90470 8.36589 12.50976 22.17569 9.92401 8.39261 2.46479  carrierUS carrierVX carrierWN carrierYV distance month hour  8.90832 -0.48704 -0.37823 6.45895 -0.09116 0.21378 -0.06230 |

1. Apply 10-fold cross-validation, to the model. [2 points]

Your input and output code goes here:

cv <- crossv\_kfold(flights, k=10)

models <-

  map(cv$train, ~lm(arr\_delay ~ origin + dep\_delay + air\_time + carrier + distance + month + hour, data=.))

models[[6]]

Table 22. Model summary of the 6th cross-validation fold.

|  |
| --- |
| Call:  lm(formula = arr\_delay ~ origin + dep\_delay + air\_time + carrier +  distance + month + hour, data = .)  Coefficients:  (Intercept) originJFK originLGA dep\_delay air\_time carrierAA carrierAS carrierB6  -22.59318 -0.82471 -0.57962 1.02308 0.71074 2.41317 -6.62903 8.96045  carrierDL carrierEV carrierF9 carrierFL carrierHA carrierMQ carrierOO carrierUA  4.26356 4.88954 8.38487 12.55626 21.73103 9.92370 7.26575 2.39520  carrierUS carrierVX carrierWN carrierYV distance month hour  8.92508 -0.52682 -0.49329 6.48192 -0.09113 0.21334 -0.06158 |

1. Compute the average mean-square prediction error for arrival delay using this model. [3 points]

Your input and output code goes here:

get\_pred <- function(model, test\_data) {

  data <- as.data.frame(test\_data)

  pred <- add\_predictions(data, model)

  return(pred)

}

predictions <- map2\_df(models, cv$test, get\_pred, .id = "Run")

MSE <- predictions %>%

  drop\_na() %>%

  group\_by(Run) %>%

  summarise(MSE = mean( (arr\_delay - pred)^2))

mean(MSE$MSE)

Table 23.Mean square error for the 10-fold cross-validated final model.

|  |
| --- |
| [1] 233.7122 |