Challenge

AutosRUS Analysis

Alberto Valdez

October 13, 2022

Contents

1 Part 1: Predict MPG

This document is a complementary writeup done in the stye of "literate programming" for writing the R code and gathering the results.

For the report, go to `./readme.org` or README.md.

```
library(dplyr)
library(ggplot2)
library(tidyverse)
```

1.1 Import Data

```
mpgcar <-
  read.csv(
    'MechaCar_mpg.csv',
```

```
    check.names = F,
    stringsAsFactors = F
)
head(mpgcar)
```

vehicle_length	vehicle_weight	spoiler_angle	ground_clearance	AWD	mpg
14.69709536	6407.94647	48.78998258	14.64098303	1	49.04918045
12.53420597	5182.080571	90	14.36667939	1	36.76606309
20	8337.981208	78.63232282	12.25371141	0	80
13.42848546	9419.670939	55.93903153	12.98935921	1	18.9414895
15.44997974	3772.666826	26.12816424	15.10396274	1	63.82456769
14.45356979	7286.594508	30.58567612	13.10695343	0	48.54267684

1.2 Linear regression

```
lm(
  mpg ~ vehicle_length +
  vehicle_weight +
  spoiler_angle +
  ground_clearance +
  AWD,
  data = mpgcar
)
```

Call:
lm(formula = mpg ~ vehicle_length + vehicle_weight + spoiler_angle +
ground_clearance + AWD, data = mpgcar)

Coefficients:
(Intercept) vehicle_length vehicle_weight spoiler_angle
-1.040e+02 6.267e+00 1.245e-03 6.877e-02
ground_clearance AWD
3.546e+00 -3.411e+00

```
summary(
  lm(
    mpg ~ vehicle_length +
    vehicle_weight +
    spoiler_angle +
```

```
    ground_clearance +
    AWD,
    data = mpgcar
  )
)
```

Call:
lm(formula = mpg ~ vehicle_length + vehicle_weight + spoiler_angle + ground_clearance + AWD, data = mpgcar)

Residuals:
Min 1Q Median 3Q Max
-19.4701 -4.4994 -0.0692 5.4433 18.5849

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.040e+02 1.585e+01 -6.559 5.08e-08 ***
vehicle_length 6.267e+00 6.553e-01 9.563 2.60e-12 ***
vehicle_weight 1.245e-03 6.890e-04 1.807 0.0776 .
spoiler_angle 6.877e-02 6.653e-02 1.034 0.3069
ground_clearance 3.546e+00 5.412e-01 6.551 5.21e-08 ***
AWD -3.411e+00 2.535e+00 -1.346 0.1852
—
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.774 on 44 degrees of freedom
Multiple R-squared: 0.7149, Adjusted R-squared: 0.6825
F-statistic: 22.07 on 5 and 44 DF, p-value: 5.35e-11

2 Part 2: Trip Analysis Visualization

```
coildata <-
   read.csv(
    'Suspension_Coil.csv',
    check.names = F,
    stringsAsFactors = F
)
head(coildata)
```

VehicleID	Manufacturing_Lot	PSI
V40858	Lot1	1499
V40607	Lot1	1500
V31443	Lot1	1500
V6004	Lot1	1500
V7000	Lot1	1501
V17344	Lot1	1501

Write an RScript that creates a total_summary dataframe using the summarize() function to get the mean, median, variance, and standard deviation of the suspension coil's PSI column.

```
total_summary <-
  coildata %>%
  summarize(
    Mean=mean(PSI),
    Median=median(PSI),
    Variance=var(PSI),
    SD=sd(PSI)
)
```

Mean	Median	Variance	SD
1498.78	1500	62.2935570469799	7.89262675203762

```
lot_summary <-
  coildata %>%
  group_by(Manufacturing_Lot) %>%
  summarize(
    Mean=mean(PSI),
    Median=median(PSI),
    Variance=var(PSI),
    SD=sd(PSI),
    .groups='keep'
  )
```

Manufacturing_Lot	Mean	Median	Variance	SD
Lot1	1500	1500	0.979591836734694	0.989743318610787
Lot2	1500.2	1500	7.46938775510204	2.73301806710128
Lot3	1496.14	1498.5	170.28612244898	13.0493724925369

3 Part 3: T-Tests on Suspension Coils

In your MechaCarChallenge.RScript, write an RScript using the t.test() function to determine if the PSI across all manufacturing lots is statistically different from the population mean of 1,500 pounds per square inch.

```
t.test(
  coildata$PSI,
  mu=mean(coildata$PSI)
)
```

One Sample t-test

data: coildata$PSI
t = 0, df = 149, p-value = 1
alternative hypothesis: true mean is not equal to 1498.78
95 percent confidence interval:
1497.507 1500.053
sample estimates:
mean of x
1498.78

Next, write three more RScripts in your MechaCarChallenge.RScript using the t.test() function and its subset() argument to determine if the PSI for each manufacturing lot is statistically different from the population mean of 1,500 pounds per square inch.

```
t.test(
  subset(
    coildata$PSI,
    coildata$Manufacturing_Lot == "Lot1"
  ),
  mu=mean(coildata$PSI)
)
```

One Sample t-test

data: subset(coildata$PSI, coildata$Manufacturing_Lot == “Lot1”)
t = 8.7161, df = 49, p-value = 1.568e-11

alternative hypothesis: true mean is not equal to 1498.78
95 percent confidence interval:
1499.719 1500.281
sample estimates:
mean of x
1500

```
t.test(
  subset(
    coildata$PSI,
    coildata$Manufacturing_Lot == "Lot2"
  ),
  mu=mean(coildata$PSI)
)
```

One Sample t-test

data: subset(coildata$PSI, coildata$Manufacturing_Lot == “Lot2”)
t = 3.6739, df = 49, p-value = 0.0005911
alternative hypothesis: true mean is not equal to 1498.78
95 percent confidence interval:
1499.423 1500.977
sample estimates:
mean of x
1500.2

```
t.test(
  subset(
    coildata$PSI,
    coildata$Manufacturing_Lot == "Lot3"
  ),
  mu=mean(coildata$PSI)
)
```

One Sample t-test

data: subset(coildata$PSI, coildata$Manufacturing_Lot == “Lot3”)
t = -1.4305, df = 49, p-value = 0.1589

alternative hypothesis: true mean is not equal to 1498.78
95 percent confidence interval:
1492.431 1499.849
sample estimates:
mean of x
1496.14

4 Complete script

Complete code for MechaCharChallenge.

```
library(dplyr)
library(ggplot2)
library(tidyverse)
mpgcar <-
  read.csv(
    'MechaCar_mpg.csv',
    check.names = F,
    stringsAsFactors = F
)
head(mpgcar)
lm(
  mpg ~ vehicle_length +
  vehicle_weight +
  spoiler_angle +
  ground_clearance +
  AWD,
  data = mpgcar
)
summary(
  lm(
    mpg ~ vehicle_length +
    vehicle_weight +
    spoiler_angle +
    ground_clearance +
    AWD,
    data = mpgcar
  )
```

```
)
coildata <-
   read.csv(
    'Suspension_Coil.csv',
    check.names = F,
    stringsAsFactors = F
)
head(coildata)
total_summary <-
  coildata %>%
  summarize(
    Mean=mean(PSI),
    Median=median(PSI),
    Variance=var(PSI),
    SD=sd(PSI)
)
lot_summary <-
  coildata %>%
  group_by(Manufacturing_Lot) %>%
  summarize(
    Mean=mean(PSI),
    Median=median(PSI),
    Variance=var(PSI),
    SD=sd(PSI),
    .groups='keep'
  )
t.test(
  coildata$PSI,
  mu=mean(coildata$PSI)
)
t.test(
  subset(
    coildata$PSI,
    coildata$Manufacturing_Lot == "Lot1"
  ),
  mu=mean(coildata$PSI)
)
t.test(
  subset(
    coildata$PSI,
```

```
    coildata$Manufacturing_Lot == "Lot2"
  ),
  mu=mean(coildata$PSI)
)
t.test(
  subset(
    coildata$PSI,
    coildata$Manufacturing_Lot == "Lot3"
  ),
  mu=mean(coildata$PSI)
)

<<rscript>>
```