# Ryan Gallagher

# BIOS-04232 HW1

# January 31st, 2023

# Due 2/7/2023

air.csv includes a Civil Aeronautics Board report from SAS for regression.

The data concerns important predictors in predicting the cost of providing air service.

# The variables:

CPM: cost per passenger mile (cents)

UTL: average hours per day use of aircraft (hrs)

ASL: average length of nonstop legs of flights (1000 miles)

SPA: average number of seats per aircraft (100 seats)

ALF: average load factor (% of seats occupied by passengers)

CPM is the response varibale. ALL OTHERS ARE PREDICTORS.

air = read.csv("/Users/ryangallagher/Desktop/MedicalCollegeofWisconsin/BIOS\_04232\_MM2/HW1-RyanGallagher/air.csv")

#————————- HOMEWORK QUETIONS ——————————-  
#—— Q1. ————-  
#Q1: Fit a linear regression model using SAS (PROC REG - see .sas file) and R(lm). State your linear regression model with assumptions. Are all predictors significant at a significance level ? Interpret the coefficient of UTL.

lmAir = lm(CPM~UTL+ASL+SPA+ALF, data=air)  
summary(lmAir)

##   
## Call:  
## lm(formula = CPM ~ UTL + ASL + SPA + ALF, data = air)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.09885 -0.20042 -0.01266 0.21688 0.86564   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.59553 0.90278 9.521 2.81e-10 \*\*\*  
## UTL -0.21282 0.06509 -3.270 0.00285 \*\*   
## ASL 0.33277 0.18133 1.835 0.07713 .   
## SPA -4.95030 1.21695 -4.068 0.00035 \*\*\*  
## ALF -7.21137 1.32056 -5.461 7.88e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3945 on 28 degrees of freedom  
## Multiple R-squared: 0.6013, Adjusted R-squared: 0.5443   
## F-statistic: 10.56 on 4 and 28 DF, p-value: 2.417e-05

# Under the assumptions of multiple linear regression:

1. There must be a linear relationship between the dependent variable and each independent variable.
2. Normally distributed residuals
3. No Multicollinearity - i.e there is no/little interaction between dependent variables
4. Homoscedasticity

Under these assumptions: CPM = -0.212(UTL) + 0.3327(ASL) - 4.95(SPA) - 7.211 (AFL) + 8.595

Where only the variable ASL is NOT a significant predictor, thus a change in ASL is not associated with change in CPM.

Interpret coefficient UTL: While all other independent variables stay constant, a one unit increase in UTL decreases CPM by 0.212. That is, for every one hour per day of aircraft usage, there is a 0.212 decrease in cost per mile (in cents).

#——-  
#Q2  
#———————————-  
#———————————–

#Q2: Obtain B^hat, its standard errors, and 95% CIs using matrix calculation in R.

xAir = cbind(rep(1,dim(air)[1]),as.matrix(air[,c(1:4)])) #These are the independent variables  
yAir=air$CPM; nAir=dim(air)[1]  
betahatAir = solve(t(xAir)%\*%xAir)%\*%t(xAir)%\*%yAir #This matches  
  
yhatAir = xAir %\*% betahatAir  
SSEair = sum((yAir - yhatAir)^2)  
SSRair = sum((yhatAir - mean(yAir))^2)  
dfAir = nAir-4-1  
sigma2hatAir = SSEair/dfAir  
varbetahatAir = sigma2hatAir \* solve(t(xAir)%\*%xAir)  
std.errorAir = sqrt(diag(varbetahatAir)) #This matches  
tstatisticAir = abs(betahatAir)/std.errorAir  
tstatisticAir

## [,1]  
## 9.521221  
## UTL 3.269741  
## ASL 1.835122  
## SPA 4.067785  
## ALF 5.460833

MSEair = SSRair / 4  
MSRair = SSEair / 28  
  
Fair = MSEair / MSRair  
Fair

## [1] 10.55603

#CI of betahat -> 1.96 from 95% CI  
# beta\_i +- 1.96\*s  
CIu = betahatAir+1.96\*std.errorAir  
CIl = betahatAir-1.96\*std.errorAir  
cbind(CIl,CIu)

## [,1] [,2]  
## 6.82608512 10.36496498  
## UTL -0.34038511 -0.08524634  
## ASL -0.02264457 0.68818243  
## SPA -7.33552810 -2.56507464  
## ALF -9.79967663 -4.62306986

#——————–  
#Q3  
#——————–

H0: CPM = B\_0 + B\_1*UTL + B\_3*SPA + B\_4*ALF + e H1: CPM = B\_0 + B\_1*UTL + B\_2*ASL + B\_3*SPA + B\_4\*ALF + e

Fit H0 and H1 using SAS. Perform the t-test and the F-test (by hand) using the SAS outputs from fitting H0 and H1

From SAS:

H0: CPM = 7.721 - 0.1385*UTL - 3.503*SPA - 6.203*ALF*  
*H1: CPM = 8.595 - 0.2128*UTL + 0.333*ASL - 4.950*SPA - 7.2113\*ALF

t-test and F-test for H0:

lmH0 = lm(CPM~UTL+SPA+ALF, data=air)  
summary(lmH0)

##   
## Call:  
## lm(formula = CPM ~ UTL + SPA + ALF, data = air)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.05659 -0.13973 -0.04724 0.13874 0.86271   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.72152 0.79762 9.681 1.37e-10 \*\*\*  
## UTL -0.13850 0.05299 -2.614 0.01406 \*   
## SPA -3.50357 0.96417 -3.634 0.00107 \*\*   
## ALF -6.20314 1.24891 -4.967 2.78e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4103 on 29 degrees of freedom  
## Multiple R-squared: 0.5533, Adjusted R-squared: 0.5071   
## F-statistic: 11.97 on 3 and 29 DF, p-value: 2.828e-05

xH0 = cbind(rep(1,dim(air)[1]),as.matrix(air[,c(1,3,4)]))  
yH0 = air$CPM  
nH0=dim(air)[1]  
betahatH0 = solve(t(xH0)%\*%xH0)%\*%t(xH0)%\*%yH0 #matches SAS  
  
yhatH0 = xH0 %\*% betahatH0  
SSEH0 = sum((yH0 - yhatH0)^2)  
SSRH0 = sum((yhatH0 - mean(yH0))^2)  
dfH0 = nH0-3-1  
sigma2hatH0 = SSEH0/dfH0  
varbetahatH0 = sigma2hatH0 \* solve(t(xH0)%\*%xH0)  
std.errorH0 = sqrt(diag(varbetahatH0))  
tstatisticH0 = abs(betahatH0)/std.errorH0 # matches SAS  
tstatisticH0

## [,1]  
## 9.680701  
## UTL 2.613544  
## SPA 3.633746  
## ALF 4.966854

MSRH0 = SSRH0/3  
MSEH0 = SSEH0 / 29  
  
FH0Q3 = MSRH0/MSEH0  
FH0Q3

## [1] 11.9745

Testing Models

FQ3 = ((SSEH0 - SSEair)/1)/(SSEair/(nH0-5)) #matches SAS  
FQ3

## [1] 3.367672

FP\_valQ3 = 1-pf(FQ3,1,nH0-5)  
FP\_valQ3

## [1] 0.0771321

# IMPORTANT - F = ((SSERM - SSE)/2) / (SSE/n-3) IS AN F TEST FOR BOTH MODELS TO SEE IF THEYRE SIMILAR

# SSERM = SUMS OF SQUARES ERROR RESTRICTED MODEL

# HERE, SSE = SUMS OF SQUARES ERROR FOR UNRESTRICTED MODEL (FULL MODEL)

# F = ((RRSS - URSS) / q) / (URSS / (N-k))

# q is number of restrictions, N is total cases, k is number of betas

# Interpretation from Duke Website - If the p\_val > alpha, then the FIRST model is statistically better than the second. If the p\_value < alpha, then the SECOND model is significantly better than the first.

t-test and F-test for:  
H1

tstatisticAir #This matches SAS

## [,1]  
## 9.521221  
## UTL 3.269741  
## ASL 1.835122  
## SPA 4.067785  
## ALF 5.460833

Fair #This matches SAS

## [1] 10.55603

H0

tstatisticH0 #matches SAS

## [,1]  
## 9.680701  
## UTL 2.613544  
## SPA 3.633746  
## ALF 4.966854

FH0Q3 #matches SAS

## [1] 11.9745

Compare P\_values:

H1

1-pt(tstatisticAir,4)

## [,1]  
## 0.0003396792  
## UTL 0.0153983423  
## ASL 0.0701953028  
## SPA 0.0076253458  
## ALF 0.0027336169

#F = 10.56  
#FP\_val = <.0001

H0

1-pt(tstatisticH0,3,lower.tail=TRUE)

## [,1]  
## 0.001170265  
## UTL 0.039720939  
## SPA 0.017949448  
## ALF 0.007837745

#F = 11.97  
#FP\_val = <.0001

Both models give very similar F-test p\_values. Both tests yield p\_values from the t-test that are also similar. However, when running the F test to compare both models, we get a p\_value of 0.07713. This is confirmed in SAS. Since this value is greater than 0.05, we interpret that the first model, HO, is a statistically better model than H0.  
#———–  
#———–  
Q4  
#———–  
#———-

Using the SAS Output here:

H0 / Resitricted Model:

xH04 = cbind(rep(1,dim(air)[1]),as.matrix(air[,1]))  
yH04 = air$CPM  
matH04 = matrix(nrow=2)  
betahatH04=matH04  
betahatH04[1,1]=4.413442153  
betahatH04[2,1]=-0.150017698  
  
st.errorH04 = matH04  
st.errorH04[1,1] = 0.59393100  
st.errorH04[2,1] = 0.06723785  
  
yhatH04 = xH04 %\*% betahatH04  
SSEH04 = sum((yH04 - yhatH04)^2)  
  
tstatisticH04 = abs(betahatH04)/st.errorH04  
tstatisticH04

## [,1]  
## [1,] 7.43090  
## [2,] 2.23115

H1 / Unrestricted Model:

xH14 = cbind(rep(1,dim(air)[1]),as.matrix(air[,c(1:3)]))  
yH14 = air$CPM  
matH14 = matrix(nrow=4)  
betahatH14=matH14  
betahatH14[1,1]=4.343927044  
betahatH14[2,1]=-0.107015  
betahatH14[3,1]=-0.0792085  
betahatH14[4,1]=-1.0294664  
  
st.errorH14 = matH14  
st.errorH14[1,1] = 0.64528874  
st.errorH14[2,1] = 0.08773753  
st.errorH14[3,1] = 0.23283564  
st.errorH14[4,1] = 1.38742620  
  
yhatH14 = xH14 %\*% betahatH14  
SSEH14 = sum((yH14 - yhatH14)^2)  
  
tstatisticH14 = abs(betahatH14)/st.errorH14  
  
FQ4 = ((SSEH04 - SSEH14)/2)/(SSEH14/(nH0-4)) #matches SAS  
FQ4 #matches SAS proc reg

## [1] 0.6738307

FP\_valQ4 = 1-pf(FQ4,2,nH0-4)  
FP\_valQ4 #matches SAS proc reg

## [1] 0.5175534

In this problem, I determined the p\_values for the t-statistics and F-statistic from each hypothesis and compared them, Aftewards, I performed the F-test to compare both hypothesis, and compared it’s p\_value to the SAS output of PROC REG. In the end, we found an F-statistic = 0.67 (which matched the SAS output) and a p\_value=0.5176 (matches the SAS output.) Since this p\_value > 0.05, we interpret that H0 is a statistically better model than H1.

#————  
#—-  Q5  
#—  
# ———–

The Covariance of Estimates is found in the SAS file. To test H0: B3 = B4 vs. H1: B3 != B4, we find the following t-statistic: t = (B3 - B4) / SE(B3-B4)

SE(B3 - B4) = sqrt(var(B3) + var(B4) - 2\*cov(B3,B4)) Where var(B3) and var(B4) are diagonal elements of the covariate matrix and Cov(B3,B4) is the (B3,B4) entry of the covariate matrix

From SAS:

B3 = -4.95030  
B4 = -7.21137  
var.B3 = 1.480973  
var.B4 = 1.743886  
cov.B3B4 = 0.948153  
  
SE.B3B4 = sqrt(var.B3+var.B4 - 2\*cov.B3B4)  
  
tstat.Q5 = (B3-B4) / SE.B3B4  
pt(tstat.Q5,4) #=0.06066149

## [1] 0.9393385

Getting SSE for reduced model

airx5 = air  
airx5$SPApALF = airx5$SPA+airx5$ALF  
x5 = cbind(rep(1,dim(airx5)[1]),as.matrix(airx5[,c(1,2,7)]))  
y5 = air$CPM  
nH0=dim(air)[1]  
betahatx5 = solve(t(x5)%\*%x5)%\*%t(x5)%\*%y5 #matches SAS  
  
yhatx5 = x5 %\*% betahatx5  
SSEx5 = sum((y5 - yhatx5)^2)  
SSRx5 = sum((yhatx5 - mean(y5))^2)  
dfx5 = nH0-3-1  
sigma2hatx5 = SSEx5/dfx5  
varbetahatx5 = sigma2hatx5 \* solve(t(x5)%\*%x5)  
std.errorx5 = sqrt(diag(varbetahatx5))  
tstatisticx5 = abs(betahatx5)/std.errorx5 # matches SAS  
tstatisticx5

## [,1]  
## 9.013800  
## UTL 2.979178  
## ASL 2.186710  
## SPApALF 4.964818

#Comparing the two models, I'll use the SSE values from SAS  
SSEQ5FM = 4.3575  
SSEQ5RM = SSEx5 #=4.956387  
  
fstatQ5 = ((SSEx5 - SSEQ5FM) / 1) / (SSEQ5FM/(nH0-4)) #=3.98571 - not exactly same as sas, close  
1-pf(fstatQ5,1,28) #=0.059534 - bascially the same as sas

## [1] 0.05569444

Conclusion - since our pval > 0.05, we fail to reject H0 and conclude that the reduced model is better than the model where beta3=beta4.

#—————–  
#——  
# Q6  
#——-  
#—————–

want to test H0: beta3+2\*beta4 = 17  
I found:  
y = b0 +x1b1+x2b2+(x4-2x3)b4 -17x3+e which is y+17x3 = b0 + x1b1 + x2b2 + (x4-2x3)b4 + e

# Full Model  
SSEair

## [1] 4.357519

std.errorAir

## UTL ASL SPA ALF   
## 0.90277548 0.06508642 0.18133342 1.21695241 1.32056295

tstatisticAir

## [,1]  
## 9.521221  
## UTL 3.269741  
## ASL 1.835122  
## SPA 4.067785  
## ALF 5.460833

Fair

## [1] 10.55603

# Reduced Model  
airx6 = air  
airx6$ALFm2SPA = airx6$ALF - (2\*airx6$SPA)  
airx6$CPMpSPA = airx6$CPM + (17\*airx6$SPA)  
x6 = cbind(rep(1,dim(airx6)[1]),as.matrix(airx6[,c(1,2,7)]))  
y6 = airx6$CPMpSPA  
nH0 = dim(airx6)[1]  
betahatx6=solve(t(x6)%\*%x6)%\*%t(x6)%\*%y6  
  
yhatx6 = x6 %\*% betahatx6  
SSEx6 = sum((y6-yhatx6)^2)  
SSRx6 = sum((yhatx6 - mean(y6))^2)  
dfx6 = nH0-3-1  
sigma2hatx6 = SSEx6/dfx6  
varbetahatx6 = sigma2hatx6 \* solve(t(x6)%\*%x6)  
std.errorx6 = sqrt(diag(varbetahatx6))  
tstatx6 = abs(betahatx6)/std.errorx6  
  
#P\_val for reduced model   
tstatx6 #Matches SAS

## [,1]  
## 17.573600  
## UTL 3.248665  
## ASL 1.761401  
## ALFm2SPA 17.294974

2\*(1-pt(tstatx6,29)) #Matches SAS

## [,1]  
## 0.000000000  
## UTL 0.002929584  
## ASL 0.088712138  
## ALFm2SPA 0.000000000

#F-test comparison  
  
fstatQ6 = ((SSEx6 - SSEair) / 1) / (SSEair/(nH0-3)) #=0.4925, which is not exactly what SAS outputs but is close  
1-pf(fstatQ6,1,28) #=0.5109 which is not exactly what sas says but close

## [1] 0.4885749

Conclusion - since our pval > 0.05, we fail to reject H0 and conclude that the reduced model is better than the model where beta3+2\*beta4 = -17