**STATA (For Regression):**

```
regress passengers fare fare_lg fare_low large_ms lf_ms nsmiles year quarter

avplot fare                 // Partial regression plot for fare
avplot fare_lg              // Partial regression plot for fare_lg
avplot fare_low             // Partial regression plot for fare_low
avplot large_ms             // Partial regression plot for large_ms
avplot lf_ms                // Partial regression plot for lf_ms
avplot nsmiles              // Partial regression plot for nsmiles
avplot year                 // Partial regression plot for year
avplot quarter              // Partial regression plot for quarter
```

**R (For Random Forest):**

```
install.packages("randomForest")
install.packages("tidyr")
install.packages("skimr")
install.packages("ggplot2")
install.packages("skimr")
install.packages("caret")
install.packages("pROC")
install.packages("missForest")



rm(list=ls())
set.seed(6666)
RFDATA <- read.csv("~/Downloads/businessss.csv")
RFDATA <- na.omit(RFDATA)

set.seed(6666)
SDF <- RFDATA[sample(nrow(RFDATA), 2000), ]
```

```
SDF %>% colnames()

RFDATA <- SDF

RFDATA %>% head()
RFDATA %>% colnames()

train <- createDataPartition(y=RFDATA$passengers,p=0.8,list = F)
traindata <- RFDATA[train,]
testdata <- RFDATA[-train,]

colnames(RFDATA)

form_reg <- as.formula(
  paste0(
    "passengers ~",
    paste(colnames(traindata)[1:8],collapse = "+")
  )
)
form_reg

set.seed(6666)
fit_rf_reg <- randomForest(
  form_reg,
  data=traindata,
  ntree=600,
  mtry=3,
  importance=T)
fit_rf_reg
plot(fit_rf_reg)
importance(fit_rf_reg)
```

```
varImpPlot(fit_rf_reg,main="importance")

importance <- importance(fit_rf_reg)

importance_df <- as.data.frame(importance)
importance_df$Variable <- row.names(importance_df)
rownames(importance_df) <- NULL

importance_df$IncMSE <- abs(importance_df$`%IncMSE`)

importance_df <- importance_df[order(-importance_df$IncMSE), ]

ggplot(importance_df, aes(y = reorder(Variable, IncMSE), x = IncMSE)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Variable Importance", x = "% Increase in MSE", y = "Variable") +
  theme_minimal() +
  theme(axis.text.y = element_text(angle = 0, hjust = 1)) +    geom_text(aes(label =
round(IncMSE, 2)),            vjust = 0,            hjust = 1)

ggplot(importance_df, aes(x = IncNodePurity, y = reorder(Variable, IncNodePurity))) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Variable Importance", x = "IncNodePurity ", y = "Variable") +
  theme_minimal() +
  theme(axis.text.y = element_text(angle = 0, hjust = 1)) +    geom_text(aes(label =
round(IncNodePurity, 2)),            vjust = 0,            hjust = 1)



#validation
trainpred <- predict(fit_rf_reg,newdata = traindata)
trainpred
defaultSummary(data.frame(obs=traindata$passengers,pred=trainpred))
```

```
plot(
  x=traindata$passengers,
  y=trainpred,
  xlab="actual",
  ylab="prediction",
  main="random forest comparasion RFDATA",
  sub="train"
)
trainlinmod <- lm(trainpred ~ traindata$passengers)
abline(trainlinmod, col="blue",lwd="2.5",lty="solid")
abline(a=0,b=1,col="red",lwd="2.5",lty="dashed")

legend("topleft",
       legend=c("mode","base"),
       col=c("blue","red"),
       lwd=2.5,
       lty=c("solid","dashed"))

testpred <- predict(fit_rf_reg,newdata=testdata)
testpred
defaultSummary(data.frame(obs=testdata$passengers,pred=testpred))
plot(
  x=testdata$passengers,
  y=testpred,
  xlab="actual",
  ylab="prediction",
  main="random forest comparasion RFDATA",
  sub="test"
)
```

```
testlinmod <- lm(testpred ~ testdata$passengers)
abline(testlinmod, col="blue",lwd="2.5",lty="solid")
abline(a=0,b=1,col="red",lwd="2.5",lty="dashed")
legend("topleft",
       legend=c("mode","base"),
       col=c("blue","red"),
       lwd=2.5,
       lty=c("solid","dashed"))
```

```
predresult <-
  data.frame(obs=c(traindata$RFDATAv,testdata$RFDATAv),
             pred=c(trainpred,testpred),
             group=c(rep("train",length(trainpred)),
                     rep("test",length(testpred))))

ggplot(predresult,
       aes(x=obs,y=pred,fill=group,colour=group))+
  geom_point(shape=21,size=3)+geom_smooth(method="lm",se=F,size=1.2)+
  geom_abline(intercept = 0,slope=1,size=1.2)+
  theme(legend.position="bottom")
```

**References to External Source Use:**

**Random Forest:**


1. https://www.bilibili.com/video/BV1Ag4y157rz/?spm_id_from=333.337.search-card.all.click&vd_source=c1a4c9e0b75881358f373e435dea1e91
2. https://www.bilibili.com/video/BV1Ag4y157rz/?spm_id_from=333.337.search-card.all.click

3. https://www.lianxh.cn/details/532.html
4. **CHATGPT 4o:** This was used specifically for the interpretation part on variable importance – %Increase in MSE and IncNodePurity