library(shinydashboard)

library(caret)

library(ROCR)

library(randomForest)

set.seed(13)

shinyServer(function(input,output){

#========================================Output\_of\_Routewise\_Analytics=================================

# output$Routewise\_Cancellation\_Reasons <- renderTable({

# subset(Delayflightsreasonvise,Delayflightsreasonvise$AIRLINE==input$AIRLINE & Delayflightsreasonvise$ORIGIN\_AIRPORT==input$ORIGIN\_AIRPORT & Delayflightsreasonvise$DESTINATION\_AIRPORT==input$DESTINATION\_AIRPORT)

#})

#==========================================output\_of\_Airlinewise\_Analytics==============================

output$Airlines\_Full\_Details <- renderTable({

subset(Delayflightsreasonvise, Delayflightsreasonvise$AIRLINE==input$Airlinename)

})

#================which airline to choose========

output$airlinechoose <- renderPlotly({

Delayflightsreasonvise$DayOfWeek <- Delayflightsreasonvise$DayOfWeek

Delayflightsreasonvise$Month <- Delayflightsreasonvise$Month

# Define levels

Delayflightsreasonvise$AIRLINE <- factor(Delayflightsreasonvise$AIRLINE)

Delayflightsreasonvise$Year <- factor(Delayflightsreasonvise$Year)

Delayflightsreasonvise$Month <- factor(Delayflightsreasonvise$Month)

Delayflightsreasonvise$ORIGIN\_AIRPORT <- factor(Delayflightsreasonvise$ORIGIN\_AIRPORT)

Delayflightsreasonvise$DESTINATION\_AIRPORT <- factor(Delayflightsreasonvise$DESTINATION\_AIRPORT)

# Define numbers by actual days and months.

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 1] <- 'January'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 2] <- 'February'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 3] <- 'March'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 4] <- 'April'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 5] <- 'May'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 6] <- 'June'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 7] <-'July'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 8] <- 'August'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 9] <- 'September'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 10] <- 'October'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 11] <- 'November'

Delayflightsreasonvise$Month[Delayflightsreasonvise$Month == 12] <- 'December'

p <- ggplot(Delayflightsreasonvise, aes(Month, fill = Delayflightsreasonvise$AIRLINE)) +

geom\_bar(width=0.8, position="dodge",color="black")+

labs(x = "Month", y = "Count",title = "Flight Delay Counts by Airline Carriers(By Month)", fill = "Airlines")+

theme(legend.text = element\_text(colour="purple", size=10,

face="bold"))

print(p)

ggplotly(p)

})

output$airlinechooseweek <- renderPlotly({

Delayflightsreasonvise$DayOfWeek <- Delayflightsreasonvise$DayOfWeek

Delayflightsreasonvise$Month <- Delayflightsreasonvise$Month

# Define levels

Delayflightsreasonvise$AIRLINE <- factor(Delayflightsreasonvise$AIRLINE)

Delayflightsreasonvise$Year <- factor(Delayflightsreasonvise$Year)

Delayflightsreasonvise$Month <- factor(Delayflightsreasonvise$Month)

Delayflightsreasonvise$ORIGIN\_AIRPORT <- factor(Delayflightsreasonvise$ORIGIN\_AIRPORT)

Delayflightsreasonvise$DESTINATION\_AIRPORT <- factor(Delayflightsreasonvise$DESTINATION\_AIRPORT)

# Define numbers by actual days and months.

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 1] <- 'Monday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 2] <- 'Tuesday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 3] <- 'Wednesday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 4] <- 'Thursday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 5] <- 'Friday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 6] <- 'Saturday'

Delayflightsreasonvise$DayOfWeek[Delayflightsreasonvise$DayOfWeek == 7] <- 'Sunday'

p <- ggplot(Delayflightsreasonvise, aes(DayOfWeek, fill = Delayflightsreasonvise$AIRLINE)) +

geom\_bar(width=0.8, position="dodge",color="black")+

labs(x = "DayOfWeek", y = "Count",title = "Flight Delay Counts by Airline Carriers(By DayOfWeek)", fill = "Airlines")+

theme(legend.text = element\_text(colour="purple", size=10,

face="bold"))

print(p)

ggplotly(p)

})

#=====================================Cancellation\_analysis=============

output$Plotofcancel <- renderPlot({

Delayflightsreasonvise$CANCELLED[Delayflightsreasonvise$Cancelled == 0] = 'No'

Delayflightsreasonvise$CANCELLED[Delayflightsreasonvise$Cancelled == 1] = 'Yes'

qplot(factor(Delayflightsreasonvise$CANCELLED), data= Delayflightsreasonvise, geom="bar", fill=factor(Delayflightsreasonvise$CANCELLED))

})

output$plotmOfcancellationreason <- renderPlot({

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'A'] = 'Carrier'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'B'] = 'Weather'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'C'] = 'NAS'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'D'] = 'Security'

Delayflightsreasonvise %>% filter(Delayflightsreasonvise$CReason != 'N')

# Weather is the bigggest reason

CancelledSubset = subset(Delayflightsreasonvise, CReason != 'N')

ggplot(CancelledSubset,aes(Year,fill=CReason)) + geom\_bar()

})

output$plotmOfcancellationreasonwithairline <- renderPlot({

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'A'] = 'Carrier'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'B'] = 'Weather'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'C'] = 'NAS'

Delayflightsreasonvise$CReason[Delayflightsreasonvise$CReason == 'D'] = 'Security'

Delayflightsreasonvise %>% filter(Delayflightsreasonvise$CReason != 'N')

# Weather is the bigggest reason

CancelledSubset = subset(Delayflightsreasonvise, CReason != 'N')

ggplot(CancelledSubset,aes(AIRLINE,fill=CReason)) + geom\_bar()

})

#====================================Output\_of\_Departure\_Delay\_Analytics=====================

output$PlotofOnTime <- renderPlotly({

plot\_ly(Delayflightsreasonvise, x=~Delayflightsreasonvise$Year, y =~Delayflightsreasonvise$On\_Time, color =~Delayflightsreasonvise$AIRLINE, type = "bar")

})

output$plotmOfDelay <- renderPlotly({

plot\_ly(Delayflightsreasonvise, x=~Delayflightsreasonvise$AIRLINE, y =~Delayflightsreasonvise$Delay\_Time, color =~Delayflightsreasonvise$Month, type = "bar")

})

output$ plotmOfDelaybydayofweek <- renderPlotly({

plot\_ly(Delayflightsreasonvise, x=~Delayflightsreasonvise$DayOfWeek, y =~Delayflightsreasonvise$Delay\_Time, color =~Delayflightsreasonvise$AIRLINE, type = "bar")

})

#==========================================output of Airport Analytics======================

output$Airportwise\_analytics <- renderTable({

subset(Airport\_Iatacode\_Details,Airport\_Iatacode\_Details$STATE==input$STATE)

})

output$Airportwise <- renderPlotly({

Airport\_Iatacode\_Details <- plot\_ly(y=Airport\_Iatacode\_Details$CITY, x=Airport\_Iatacode\_Details$AIRPORT,color=Airport\_Iatacode\_Details$STATE,type = "scatter")

})

#=========================UA airline frequent delay reason(delay analytics)========

output$UAairlinechoose <- renderPlotly({

UAdelayreason$Total[ UAdelayreason$Total == '1'] = 'Air system delay'

UAdelayreason$Total[ UAdelayreason$Total == '2'] = 'Security delay'

UAdelayreason$Total[ UAdelayreason$Total == '3'] = 'Airline delay'

UAdelayreason$Total[ UAdelayreason$Total == '4'] = 'Late aircraft delay'

UAdelayreason$Total[ UAdelayreason$Total == '5'] = 'Weather delay'

UAdelayreason$Total[ UAdelayreason$Total == '6'] = 'Carrier delay'

UAdelayreason$Total[ UAdelayreason$Total == '7'] = 'NAS delay'

UAdelayreason$Total[ UAdelayreason$Total == '8'] = 'Baggage error'

UAdelayreason$Total[ UAdelayreason$Total == '9'] = 'Airspace congestion'

UAdelayreason$Total[ UAdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

UAdelayreason %>% filter(UAdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(UAdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================AA airline frequent delay reason(delay analytics)========

output$AAairlinechoose <- renderPlotly({

AAdelayreason$Total[ AAdelayreason$Total == '1'] = 'Air system delay'

AAdelayreason$Total[ AAdelayreason$Total == '2'] = 'Security delay'

AAdelayreason$Total[ AAdelayreason$Total == '3'] = 'Airline delay'

AAdelayreason$Total[ AAdelayreason$Total == '4'] = 'Late aircraft delay'

AAdelayreason$Total[ AAdelayreason$Total == '5'] = 'Weather delay'

AAdelayreason$Total[ AAdelayreason$Total == '6'] = 'Carrier delay'

AAdelayreason$Total[ AAdelayreason$Total == '7'] = 'NAS delay'

AAdelayreason$Total[ AAdelayreason$Total == '8'] = 'Baggage error'

AAdelayreason$Total[ AAdelayreason$Total == '9'] = 'Airspace congestion'

AAdelayreason$Total[ AAdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

AAdelayreason %>% filter(AAdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(AAdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================OH airline frequent delay reason(delay analytics)========

output$OHairlinechoose <- renderPlotly({

OHdelayreason$Total[ OHdelayreason$Total == '1'] = 'Air system delay'

OHdelayreason$Total[ OHdelayreason$Total == '2'] = 'Security delay'

OHdelayreason$Total[ OHdelayreason$Total == '3'] = 'Airline delay'

OHdelayreason$Total[ OHdelayreason$Total == '4'] = 'Late aircraft delay'

OHdelayreason$Total[ OHdelayreason$Total == '5'] = 'Weather delay'

OHdelayreason$Total[ OHdelayreason$Total == '6'] = 'Carrier delay'

OHdelayreason$Total[ OHdelayreason$Total == '7'] = 'NAS delay'

OHdelayreason$Total[ OHdelayreason$Total == '8'] = 'Baggage error'

OHdelayreason$Total[ OHdelayreason$Total == '9'] = 'Airspace congestion'

OHdelayreason$Total[ OHdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

OHdelayreason %>% filter(OHdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(OHdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================F9 airline frequent delay reason(delay analytics)========

output$F9airlinechoose <- renderPlotly({

F9delayreason$Total[ F9delayreason$Total == '1'] = 'Air system delay'

F9delayreason$Total[ F9delayreason$Total == '2'] = 'Security delay'

F9delayreason$Total[ F9delayreason$Total == '3'] = 'Airline delay'

F9delayreason$Total[ F9delayreason$Total == '4'] = 'Late aircraft delay'

F9delayreason$Total[ F9delayreason$Total == '5'] = 'Weather delay'

F9delayreason$Total[ F9delayreason$Total == '6'] = 'Carrier delay'

F9delayreason$Total[ F9delayreason$Total == '7'] = 'NAS delay'

F9delayreason$Total[ F9delayreason$Total == '8'] = 'Baggage error'

F9delayreason$Total[ F9delayreason$Total == '9'] = 'Airspace congestion'

F9delayreason$Total[ F9delayreason$Total == '10'] = 'suddenly landing due to technical problem'

F9delayreason %>% filter(F9delayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(F9delayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================B6 airline frequent delay reason(delay analytics)========

output$B6airlinechoose <- renderPlotly({

B6delayreason$Total[ B6delayreason$Total == '1'] = 'Air system delay'

B6delayreason$Total[ B6delayreason$Total == '2'] = 'Security delay'

B6delayreason$Total[ B6delayreason$Total == '3'] = 'Airline delay'

B6delayreason$Total[ B6delayreason$Total == '4'] = 'Late aircraft delay'

B6delayreason$Total[ B6delayreason$Total == '5'] = 'Weather delay'

B6delayreason$Total[ B6delayreason$Total == '6'] = 'Carrier delay'

B6delayreason$Total[ B6delayreason$Total == '7'] = 'NAS delay'

B6delayreason$Total[ B6delayreason$Total == '8'] = 'Baggage error'

B6delayreason$Total[ B6delayreason$Total == '9'] = 'Airspace congestion'

B6delayreason$Total[ B6delayreason$Total == '10'] = 'suddenly landing due to technical problem'

B6delayreason %>% filter(B6delayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(B6delayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================OO airline frequent delay reason(delay analytics)========

output$OOairlinechoose <- renderPlotly({

OOdelayreason$Total[ OOdelayreason$Total == '1'] = 'Air system delay'

OOdelayreason$Total[ OOdelayreason$Total == '2'] = 'Security delay'

OOdelayreason$Total[ OOdelayreason$Total == '3'] = 'Airline delay'

OOdelayreason$Total[ OOdelayreason$Total == '4'] = 'Late aircraft delay'

OOdelayreason$Total[ OOdelayreason$Total == '5'] = 'Weather delay'

OOdelayreason$Total[ OOdelayreason$Total == '6'] = 'Carrier delay'

OOdelayreason$Total[ OOdelayreason$Total == '7'] = 'NAS delay'

OOdelayreason$Total[ OOdelayreason$Total == '8'] = 'Baggage error'

OOdelayreason$Total[ OOdelayreason$Total == '9'] = 'Airspace congestion'

OOdelayreason$Total[ OOdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

OOdelayreason %>% filter(OOdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(OOdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================AS airline frequent delay reason(delay analytics)========

output$ASairlinechoose <- renderPlotly({

ASdelayreason$Total[ ASdelayreason$Total == '1'] = 'Air system delay'

ASdelayreason$Total[ ASdelayreason$Total == '2'] = 'Security delay'

ASdelayreason$Total[ ASdelayreason$Total == '3'] = 'Airline delay'

ASdelayreason$Total[ ASdelayreason$Total == '4'] = 'Late aircraft delay'

ASdelayreason$Total[ ASdelayreason$Total == '5'] = 'Weather delay'

ASdelayreason$Total[ ASdelayreason$Total == '6'] = 'Carrier delay'

ASdelayreason$Total[ ASdelayreason$Total == '7'] = 'NAS delay'

ASdelayreason$Total[ ASdelayreason$Total == '8'] = 'Baggage error'

ASdelayreason$Total[ ASdelayreason$Total == '9'] = 'Airspace congestion'

ASdelayreason$Total[ ASdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

ASdelayreason %>% filter(ASdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(ASdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================NK airline frequent delay reason(delay analytics)========

output$NKairlinechoose <- renderPlotly({

NKdelayreason$Total[ NKdelayreason$Total == '1'] = 'Air system delay'

NKdelayreason$Total[ NKdelayreason$Total == '2'] = 'Security delay'

NKdelayreason$Total[ NKdelayreason$Total == '3'] = 'Airline delay'

NKdelayreason$Total[ NKdelayreason$Total == '4'] = 'Late aircraft delay'

NKdelayreason$Total[ NKdelayreason$Total == '5'] = 'Weather delay'

NKdelayreason$Total[ NKdelayreason$Total == '6'] = 'Carrier delay'

NKdelayreason$Total[ NKdelayreason$Total == '7'] = 'NAS delay'

NKdelayreason$Total[ NKdelayreason$Total == '8'] = 'Baggage error'

NKdelayreason$Total[ NKdelayreason$Total == '9'] = 'Airspace congestion'

NKdelayreason$Total[ NKdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

NKdelayreason %>% filter(NKdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(NKdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================WN airline frequent delay reason(delay analytics)========

output$WNairlinechoose <- renderPlotly({

WNdelayreason$Total[ WNdelayreason$Total == '1'] = 'Air system delay'

WNdelayreason$Total[ WNdelayreason$Total == '2'] = 'Security delay'

WNdelayreason$Total[ WNdelayreason$Total == '3'] = 'Airline delay'

WNdelayreason$Total[ WNdelayreason$Total == '4'] = 'Late aircraft delay'

WNdelayreason$Total[ WNdelayreason$Total == '5'] = 'Weather delay'

WNdelayreason$Total[ WNdelayreason$Total == '6'] = 'Carrier delay'

WNdelayreason$Total[ WNdelayreason$Total == '7'] = 'NAS delay'

WNdelayreason$Total[ WNdelayreason$Total == '8'] = 'Baggage error'

WNdelayreason$Total[ WNdelayreason$Total == '9'] = 'Airspace congestion'

WNdelayreason$Total[ WNdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

WNdelayreason %>% filter(WNdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(WNdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================DL airline frequent delay reason(delay analytics)========

output$DLairlinechoose <- renderPlotly({

DLdelayreason$Total[ DLdelayreason$Total == '1'] = 'Air system delay'

DLdelayreason$Total[ DLdelayreason$Total == '2'] = 'Security delay'

DLdelayreason$Total[ DLdelayreason$Total == '3'] = 'Airline delay'

DLdelayreason$Total[ DLdelayreason$Total == '4'] = 'Late aircraft delay'

DLdelayreason$Total[ DLdelayreason$Total == '5'] = 'Weather delay'

DLdelayreason$Total[ DLdelayreason$Total == '6'] = 'Carrier delay'

DLdelayreason$Total[ DLdelayreason$Total == '7'] = 'NAS delay'

DLdelayreason$Total[ DLdelayreason$Total == '8'] = 'Baggage error'

DLdelayreason$Total[ DLdelayreason$Total == '9'] = 'Airspace congestion'

DLdelayreason$Total[ DLdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

DLdelayreason %>% filter(DLdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(DLdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================EV airline frequent delay reason(delay analytics)========

output$EVairlinechoose <- renderPlotly({

EVdelayreason$Total[ EVdelayreason$Total == '1'] = 'Air system delay'

EVdelayreason$Total[ EVdelayreason$Total == '2'] = 'Security delay'

EVdelayreason$Total[ EVdelayreason$Total == '3'] = 'Airline delay'

EVdelayreason$Total[ EVdelayreason$Total == '4'] = 'Late aircraft delay'

EVdelayreason$Total[ EVdelayreason$Total == '5'] = 'Weather delay'

EVdelayreason$Total[ EVdelayreason$Total == '6'] = 'Carrier delay'

EVdelayreason$Total[ EVdelayreason$Total == '7'] = 'NAS delay'

EVdelayreason$Total[ EVdelayreason$Total == '8'] = 'Baggage error'

EVdelayreason$Total[ EVdelayreason$Total == '9'] = 'Airspace congestion'

EVdelayreason$Total[ EVdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

EVdelayreason %>% filter(EVdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(EVdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================HA airline frequent delay reason(delay analytics)========

output$HAairlinechoose <- renderPlotly({

HAdelayreason$Total[ HAdelayreason$Total == '1'] = 'Air system delay'

HAdelayreason$Total[ HAdelayreason$Total == '2'] = 'Security delay'

HAdelayreason$Total[ HAdelayreason$Total == '3'] = 'Airline delay'

HAdelayreason$Total[ HAdelayreason$Total == '4'] = 'Late aircraft delay'

HAdelayreason$Total[ HAdelayreason$Total == '5'] = 'Weather delay'

HAdelayreason$Total[ HAdelayreason$Total == '6'] = 'Carrier delay'

HAdelayreason$Total[ HAdelayreason$Total == '7'] = 'NAS delay'

HAdelayreason$Total[ HAdelayreason$Total == '8'] = 'Baggage error'

HAdelayreason$Total[ HAdelayreason$Total == '9'] = 'Airspace congestion'

HAdelayreason$Total[ HAdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

HAdelayreason %>% filter(HAdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(HAdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================MQ airline frequent delay reason(delay analytics)========

output$MQairlinechoose <- renderPlotly({

MQdelayreason$Total[ MQdelayreason$Total == '1'] = 'Air system delay'

MQdelayreason$Total[ MQdelayreason$Total == '2'] = 'Security delay'

MQdelayreason$Total[ MQdelayreason$Total == '3'] = 'Airline delay'

MQdelayreason$Total[ MQdelayreason$Total == '4'] = 'Late aircraft delay'

MQdelayreason$Total[ MQdelayreason$Total == '5'] = 'Weather delay'

MQdelayreason$Total[ MQdelayreason$Total == '6'] = 'Carrier delay'

MQdelayreason$Total[ MQdelayreason$Total == '7'] = 'NAS delay'

MQdelayreason$Total[ MQdelayreason$Total == '8'] = 'Baggage error'

MQdelayreason$Total[ MQdelayreason$Total == '9'] = 'Airspace congestion'

MQdelayreason$Total[ MQdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

MQdelayreason %>% filter(MQdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(MQdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================VX airline frequent delay reason(delay analytics)========

output$VXairlinechoose <- renderPlotly({

VXdelayreason$Total[ VXdelayreason$Total == '1'] = 'Air system delay'

VXdelayreason$Total[ VXdelayreason$Total == '2'] = 'Security delay'

VXdelayreason$Total[ VXdelayreason$Total == '3'] = 'Airline delay'

VXdelayreason$Total[ VXdelayreason$Total == '4'] = 'Late aircraft delay'

VXdelayreason$Total[ VXdelayreason$Total == '5'] = 'Weather delay'

VXdelayreason$Total[ VXdelayreason$Total == '6'] = 'Carrier delay'

VXdelayreason$Total[ VXdelayreason$Total == '7'] = 'NAS delay'

VXdelayreason$Total[ VXdelayreason$Total == '8'] = 'Baggage error'

VXdelayreason$Total[ VXdelayreason$Total == '9'] = 'Airspace congestion'

VXdelayreason$Total[ VXdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

VXdelayreason %>% filter(VXdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(VXdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================G4 airline frequent delay reason(delay analytics)========

output$G4airlinechoose <- renderPlotly({

G4delayreason$Total[ G4delayreason$Total == '1'] = 'Air system delay'

G4delayreason$Total[ G4delayreason$Total == '2'] = 'Security delay'

G4delayreason$Total[ G4delayreason$Total == '3'] = 'Airline delay'

G4delayreason$Total[ G4delayreason$Total == '4'] = 'Late aircraft delay'

G4delayreason$Total[ G4delayreason$Total == '5'] = 'Weather delay'

G4delayreason$Total[ G4delayreason$Total == '6'] = 'Carrier delay'

G4delayreason$Total[ G4delayreason$Total == '7'] = 'NAS delay'

G4delayreason$Total[ G4delayreason$Total == '8'] = 'Baggage error'

G4delayreason$Total[ G4delayreason$Total == '9'] = 'Airspace congestion'

G4delayreason$Total[ G4delayreason$Total == '10'] = 'suddenly landing due to technical problem'

G4delayreason %>% filter(G4delayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(G4delayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================YX airline frequent delay reason(delay analytics)========

output$YXairlinechoose <- renderPlotly({

YXdelayreason$Total[ YXdelayreason$Total == '1'] = 'Air system delay'

YXdelayreason$Total[ YXdelayreason$Total == '2'] = 'Security delay'

YXdelayreason$Total[ YXdelayreason$Total == '3'] = 'Airline delay'

YXdelayreason$Total[ YXdelayreason$Total == '4'] = 'Late aircraft delay'

YXdelayreason$Total[ YXdelayreason$Total == '5'] = 'Weather delay'

YXdelayreason$Total[ YXdelayreason$Total == '6'] = 'Carrier delay'

YXdelayreason$Total[ YXdelayreason$Total == '7'] = 'NAS delay'

YXdelayreason$Total[ YXdelayreason$Total == '8'] = 'Baggage error'

YXdelayreason$Total[ YXdelayreason$Total == '9'] = 'Airspace congestion'

YXdelayreason$Total[ YXdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

YXdelayreason %>% filter(YXdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(YXdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================CO airline frequent delay reason(delay analytics)========

output$COairlinechoose <- renderPlotly({

COdelayreason$Total[ COdelayreason$Total == '1'] = 'Air system delay'

COdelayreason$Total[ COdelayreason$Total == '2'] = 'Security delay'

COdelayreason$Total[ COdelayreason$Total == '3'] = 'Airline delay'

COdelayreason$Total[ COdelayreason$Total == '4'] = 'Late aircraft delay'

COdelayreason$Total[ COdelayreason$Total == '5'] = 'Weather delay'

COdelayreason$Total[ COdelayreason$Total == '6'] = 'Carrier delay'

COdelayreason$Total[ COdelayreason$Total == '7'] = 'NAS delay'

COdelayreason$Total[ COdelayreason$Total == '8'] = 'Baggage error'

COdelayreason$Total[ COdelayreason$Total == '9'] = 'Airspace congestion'

COdelayreason$Total[ COdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

COdelayreason %>% filter(COdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(COdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=========================YV airline frequent delay reason(delay analytics)========

output$YVairlinechoose <- renderPlotly({

YVdelayreason$Total[ YVdelayreason$Total == '1'] = 'Air system delay'

YVdelayreason$Total[ YVdelayreason$Total == '2'] = 'Security delay'

YVdelayreason$Total[ YVdelayreason$Total == '3'] = 'Airline delay'

YVdelayreason$Total[ YVdelayreason$Total == '4'] = 'Late aircraft delay'

YVdelayreason$Total[ YVdelayreason$Total == '5'] = 'Weather delay'

YVdelayreason$Total[ YVdelayreason$Total == '6'] = 'Carrier delay'

YVdelayreason$Total[ YVdelayreason$Total == '7'] = 'NAS delay'

YVdelayreason$Total[ YVdelayreason$Total == '8'] = 'Baggage error'

YVdelayreason$Total[ YVdelayreason$Total == '9'] = 'Airspace congestion'

YVdelayreason$Total[ YVdelayreason$Total == '10'] = 'suddenly landing due to technical problem'

YVdelayreason %>% filter(YVdelayreason$Total != 'N')

# Weather is the bigggest reason

CancelledSubsets = subset(YVdelayreason, Total != 'N')

p <- ggplot(CancelledSubsets,aes(Month,fill=Total)) + geom\_bar()

print(p)

ggplotly(p)

})

#=================Time series===================

output$Tm <- renderPlot({

ap<- AirPassengers

ap

AP= AirPassengers$AirPassengers

class(ap)

plot(x=AirPassengers$Time,y= AP,col=c("red"))

})

#===========================Review\_analytics===============

# output$wordcloud <- renderPlot({

# setwd("E:/Project/Dataset")

# getwd()

# readLines("content.csv")

# str(readLines("content.csv"))

# text <- paste(readLines("content.csv"), collapse = " ")

# text

# text2 <- gsub(pattern = "\\W", replace =" ", text)

# text2

# text2 <- gsub(pattern = "\\d", replace =" ", text2)

# text2

# text2 <- tolower(text2)

# text2

# library(tm)

# text2 <- gsub(pattern = "\\b[A-z]\\b{1}}", replace=" ", text2)

# text2

# text2 <- stripWhitespace(text2)

# text2

# library(stringr)

# library(wordcloud)

# library(plotly)

# text2

# textbag <- str\_split(text2, pattern="\\s+")

# textbag

# class(textbag)

# textbag <- unlist(textbag)

# class(textbag)

# textbag

# str(textbag)

# getwd()

# readLines("positive-word.txt")

# poswords <- scan("positive-word.txt", what = 'character', comment.char = ";")

# str(poswords)

# negwords <- scan("negative-word.txt", what = 'character', comment.char = ";")

# str(negwords)

# match(textbag, poswords)

# match(textbag, negwords)

# posscore <- sum(!is.na(match(textbag, poswords)))

# posscore

# negscore <- sum(!is.na(match(textbag, negwords)))

# negscore

# score <- sum(!is.na(match(textbag, poswords))) - sum(!is.na(match(textbag, negwords)))

# score

# mean(score)

# hist(score)

# sd(score)

# wordcloud(textbag)

# # wordcloud(textbag, min.freq = 4)

# wordcloud(textbag,min.freq = 4, random.order = FALSE, scale = c(3, 0.5), colors = rainbow(3))

# })

#

output$sentimental<- renderPlot({

Review\_Analysis$Total[Review\_Analysis$Total == 'positive'] = 'positive'

Review\_Analysis$Total[Review\_Analysis$Total == 'negative'] = 'negative'

qplot(factor(Review\_Analysis$Total), data= Review\_Analysis, geom="bar", fill=factor(Review\_Analysis$Total))

})

output$sentimentalall <- renderPlotly({

Review\_Analysis$Total <- Review\_Analysis$Total

Review\_Analysis$Total[Review\_Analysis$Total == 0] <- 'negative'

Review\_Analysis$Total[Review\_Analysis$Total == 1] <- 'positive'

p <- ggplot(Review\_Analysis, aes(Total, fill = Review\_Analysis$`Airline name`)) +

geom\_bar(width=0.8, position="dodge",color="black")+

labs(x = "Total", y = "Count",title = "Reviews", fill = "Airlines")+

theme(legend.text = element\_text(colour="blue", size=10,

face="bold"))

print(p)

ggplotly(p)

})

#=====================================Logistic1===================

output$logisticcompare1 <- renderTable({

set.seed(123)

train\_Row =sample(1:nrow(NKreviews),nrow(NKreviews)\*0.7)

train\_data= NKreviews[train\_Row,]

test\_data= NKreviews[-train\_Row,]

model= randomForest(recommended ~ overall\_rating + seat\_comfort\_rating +cabin\_staff\_rating +food\_beverages\_rating +inflight\_entertainment\_rating +value\_money\_rating ,data=train\_data, keep.forest=TRUE,ntree=50)

print(model)

model$importance

round(importance(model),2)

varImpPlot(model)

#prediction on train data

Pred\_train=predict(model,train\_data,type ="response",norm.votes=TRUE)

#confusion matrix of train data

cm\_train=table("actual"= train\_data$recommended,"predict"=Pred\_train);

accu\_train=sum(diag(cm\_train))/sum(cm\_train)

#prediction on test data

Pred\_test=predict(model,test\_data,type ="response",norm.votes=TRUE)

#confusion matrix of train data

cm\_test=table("actual"= test\_data$recommended,"predict"=Pred\_test);

accu\_test=sum(diag(cm\_test))/sum(cm\_test)

})

})