Assignment 7

Part a

ui.R

```
library(shiny)
shinyUI(fluidPage(

  # Application title
  titlePanel("Assignment 7 - Football data"),

  # Sidebar with a slider input for number of bins
  sidebarLayout(
    sidebarPanel(
      selectInput("myIV", "Independent Variable:",
                  choices=c("Distance","ScoreDiff")),
      hr(),
      helpText("Please choose your independent variable")
    ),

    # Show a plot of the generated distribution
    mainPanel(
      verbatimTextOutput("myLogModelSummary")
    )
  )
))
```

server.R

```
library(shiny)

# Read input file
# Make sure this file is in the same folder as ui.R and server.R
kickers <- read.csv("kickers.csv")
# with attach you dont have to prepend the dataframe to the variable name
attach(kickers)

shinyServer(function(input, output) {

  output$myLogModelSummary <- renderPrint({

    # Create the formula as a string
    myFormula <- paste("Success", " ~ ", input$myIV, sep = "")
    lmfit <- glm(formula=myFormula,
                 data=kickers,
                 family=binomial)
    summary(lmfit)
  })
})
```

Part b

ui.R

```
library(shiny)

# Define UI for random distribution app ----
ui <- fluidPage(

  # Application title
  titlePanel("Assignment 7 - Football data"),

  # Sidebar with a dropdown using selectInput
  sidebarLayout(
    sidebarPanel(
      selectInput("myIV", "Independent Variable:",
                  choices=c("Distance",       "ScoreDiff")),
      hr(),
      helpText("Please choose your independent variable")

    ),

    # Main panel for displaying outputs
    mainPanel(

      # Output: Tabset with summary and plot
      tabsetPanel(type = "tabs",
                  tabPanel("Summary",
verbatimTextOutput("myLogModelSummary")),
                  tabPanel("Plot", plotOutput("myLogPlot"))
      )

    )
  )
)
```

server.R

```
library(shiny)

# Read input file
# Make sure this file is in the same folder as ui.R and server.R
kickers <- read.csv("kickers.csv")
# with attach you dont have to prepend the dataframe to the variable name
attach(kickers)

shinyServer(function(input, output) {

  # This is to prepare the summary of the output
  output$myLogModelSummary <- renderPrint({

    # Create the formula as a string
    myFormula <- paste("Success", " ~ ", input$myIV, sep = "")
    lmfit <- glm(formula=myFormula,
                 data=kickers,
                 family=binomial)
    summary(lmfit)
  })

  # This is to create a plot
  output$myLogPlot <- renderPlot({

    library(ggplot2)
    g <- ggplot(kickers, aes_string(x=input$myIV, y=Success))
    g <- g + geom_point()
    g <- g + stat_smooth(method="glm",
                  method.args=list(family="binomial"),
                  se=FALSE)
    print(g)
  })

})
```