Assignment 3 - Storytelling with Open Data

Student_name : Pragati Patidar   Student_id : s3858702

Type: Data visualisation dashboard:

URL to my published assignment :

"https://pragati22.shinyapps.io/Gender_Pay_Gap/"

All the raw code :

```
library(shiny)
library(shinydashboard)
library(ggplot2)
library(dplyr)
library(purrr)
library(plotly)
library(r2d3)
library(DT)
library(data.table)


#reading file
pg<-read.csv("Glassdoor Gender Pay Gap.csv")
pg1<-pg%>%select(BasePay,Bonus,JobTitle)

#converting into factor
pg$JobTitle<- factor(pg$JobTitle,levels = c("Warehouse Associate","Software
Engineer" ,  "Marketing Associate" ,"Sales Associate",
                                        "Financial Analyst", "IT" , "Manager",  "Data
Scientist" , "Driver","Graphic Designer" ))

job_list <-pg$JobTitle<- factor(pg$JobTitle,levels = c("Warehouse
Associate","Software Engineer" ,  "Marketing Associate" ,"Sales Associate",
                                        "Financial Analyst", "IT" , "Manager",  "Data
Scientist" , "Driver","Graphic Designer" ))

#input for department
Dept_list <- pg$Dept

#Dashboard header carrying the title of the dashboard
header <- dashboardHeader(title = "Gender paygap Dashboard", titleWidth =
350)

#Sidebar content of the dashboard
sidebar <- dashboardSidebar(
  sidebarMenu(
  menuItem("Dashboard", tabName = "dashboard", icon =
icon("dashboard")),
  menuItem(text = "About", tabName = "about", icon=icon("clipboard")),
  menuItem("Data", tabName = "data", icon=icon("database"))),

  varSelectInput("variable", "Select Variable for comparision:", pg),

  selectInput(
    inputId = "Job",
    label = "Compare Jobs and pay:",
    choices = c("Warehouse Associate","Software Engineer" ,  "Marketing
Associate" ,"Sales Associate",
          "Financial Analyst", "IT" , "Manager",  "Data Scientist" ,
"Driver","Graphic Designer"),
    selected = "DL",
    selectize = FALSE
  ),
  sidebarMenu(
    selectInput(
      inputId = "Dept",
      label = "Compare Department and pay",
      choices = c("Operations", "Management", "Sales", "Administration",
"Engineering"),
      selected = "DL",
            selectize = FALSE
        ),


        sidebarMenu(
            selectInput(
                inputId = "Education",
                label = "Compare Education and pay",
                choices = c("High School","College","PhD","Masters"),
                selected = "DL",
                selectize = FALSE
            ),

            #sidebarMenu(
               # selectInput(
                  #  inputId = "Compare Seniority and pay",
                   # label = "Seniority",
                  #  choices = c(1,2,3,4,5),
                   # selected = "DL",
                  #  selectize = FALSE
               # ) ),






            menuItem("reference link", icon = icon("send",lib='glyphicon'),
                     href = "https://www.kaggle.com/nilimajauhari/glassdoor-analyze-gender-pay-gap")
            )

        ))


body <- dashboardBody(
#frow1 <-
    fluidRow(
    valueBoxOutput("value1", width = 4)
    ,valueBoxOutput("value2",width = 4)
    ,valueBoxOutput("value3",width = 4)
),
#frow2 <-
tabItems(
    tabItem( tabName = "dashboard", fluidPage(
    fluidRow( tabBox( width = 7,
    tabPanel(
        title = "Sectors and BasePay",
        width = 6
        ,status = "primary"
        ,solidHeader = TRUE
        ,collapsible = TRUE
        ,plotlyOutput("plotline", height = "350px", width = 530)


    ),
    tabPanel(
        title = "Sectors and Bonus",
        width = 6
        ,status = "primary"
        ,solidHeader = TRUE
        ,collapsible = TRUE
        ,plotlyOutput("plotbonus", height = "350px", width = 530)


    )
    , tabPanel(
        title = "Distribution of Variables",
        width = 6
        ,status = "primary"
        ,solidHeader = TRUE
        ,collapsible = TRUE
        ,plotlyOutput("plotage", height = "350px", width = 500)
```

```
)) , tabBox( width = 5,
tabPanel(title = "Jobs and Pay",
         width = 3
         ,status = "primary"
         ,solidHeader = TRUE
         ,collapsible = TRUE
         ,plotlyOutput("plotjob", height = "300px" , width = 350)

),
tabPanel(
    title = "Department and Pay",
    width = 3
    ,status = "primary"
    ,solidHeader = TRUE
    ,collapsible = TRUE
    ,plotlyOutput("plotdept", height = "300px" , width = 350)

),
tabPanel(
    title = "Education and Pay",
    width = 3
    ,status = "primary"
    ,solidHeader = TRUE
    ,collapsible = TRUE
    ,plotlyOutput("plotedu", height = "300px", width = 350)


),

tabPanel(
    title = "Senority level and Pay",
    width = 3
    ,status = "primary"
    ,solidHeader = TRUE
    ,collapsible = TRUE
```

```
        ,plotlyOutput("plotsen", height = "300px", width = 350)
    )

    ))


   )),

    tabItem("data", fluidPage( title = "Data Table:" ,
dataTableOutput("pgtable"))),
     #initialize the ggplot2-plot object

   tabItem(tabName = "about", fluidPage( title = "About Visulisation", p("The
gender pay gap is defined as the average difference between the earnings for
male and female who are working. Generally, Women are considered to be
paid less than men. The gender pay gap is the result of many factors, here
some facors like
 job, department, education and Seniority level are Compared by this
interacive dashboard. By clciking on heading of plots, one can comapare the
pay gap and bonus of different sectors,
 Bonus and Baspay are being compared in all plots, and it can be observed
that men are getting higher pay in all sectors as comapred to women.
 Age distibution plot indicated that pay gap ih higher in the age 50 to 60
years. Bar graph compared the Pay with Job, department , education and
 seniority level at work place.Bonus is also being compared with all sectors
and Bonus pay is also higher in males. For visualising, data is taken from
Glassdoor. Details:



                    Reference of Data Source Glassdoor- Analyze Gender Pay Gap
for different job titles based on the gender,Neelima Jauhari "Glassdoor
Gender Pay Gap.csv"

 href ="https://www.kaggle.com/nilimajauhari/glassdoor-analyze-gender-
pay-gap?select=Glassdoor+Gender+Pay+Gap.csv" ")))))
```

```
#ui code
ui <- dashboardPage(title = 'Gender Pay Gap', header, sidebar, body,
skin='blue')


# creating the server functions for the dashboard
server <- function(input, output) {
    #some data manipulation to derive the values of KPI boxes
    total.salary1<-pg %>%
group_by(JobTitle,Dept,Education)%>%summarise(value = sum(BasePay))
    total.salary<-sum(total.salary1$value)

    pay <- pg %>% group_by(JobTitle) %>% summarise(value =
sum(BasePay)) %>% filter(value==max(value))
    dpt <- pg %>% group_by(Dept) %>% summarise(value = sum(BasePay))
%>% filter(value==max(value))
    #creating the valueBoxOutput content

    output$value1 <- renderValueBox({
        valueBox(
            formatC(pay$value, format="d", big.mark=',')
            ,paste('Highest Paid Job:',pay$JobTitle, width=1)
            ,icon = icon("stats",lib='glyphicon')
            ,color = "yellow")
    })
    output$value2 <- renderValueBox({
        valueBox(
            formatC( total.salary , format="d", big.mark=',')
            ,'Total pay in Thousands'
           ,icon = icon("rupee-sign",lib='glyphicon')
            ,color = "yellow")
    })
    output$value3 <- renderValueBox({
        valueBox(
            formatC(dpt$value, format="d", big.mark=',')
            ,paste('Top Department:',dpt$Dept)
        ,icon = icon("menu-hamburger",lib='glyphicon')
        ,color = "yellow")
  })
  #creating the plotlyOutput content
  output$plotjob <- renderPlotly({
    filtered<-
      pg%>%
      filter(JobTitle== input$Job)
    ggplot(data = filtered,
          aes(x=JobTitle, y=BasePay, fill=factor(Gender))) +
      geom_bar(position = "dodge", stat = "identity") + ylab("salary(in
thousands)") +
      theme(legend.position="bottom"
                        ,plot.title = element_text(size=10, face="bold")) +
      scale_fill_manual(values= c("#5F9EA0", "#E1B378"))+
      ggtitle("job and base pay") + labs(fill =
"Gender")+scale_y_continuous(labels = function(x) format(x, scientific =
FALSE))
  })
  output$plotdept <- renderPlotly({
    filteredd<-
      pg%>%
      filter(Dept== input$Dept)
    ggplotly(
      ggplot(data = filteredd,
            aes(x=Dept, y=BasePay, fill=factor(Gender))) +
        geom_bar(position = "dodge", stat = "identity") + ylab("salary (in
thousands)") +
        xlab("Department") + theme(legend.position="bottom"
                          ,plot.title = element_text(size=10, face="bold")) +
        scale_fill_manual(values= c("#5F9EA0", "#E1B378"))+
        ggtitle("Department and pay") + labs(fill =
"Gender")+scale_y_continuous(labels = function(x) format(x, scientific =
FALSE))
    ) })
  output$plotedu <- renderPlotly({
    filteredd<-
      pg%>%
      filter(Education== input$Education)
    ggplotly(
      ggplot(data = filteredd,
          aes(x=Education, y=BasePay, fill=factor(Gender))) +
        geom_bar(position = "dodge", stat = "identity") + ylab("salary (in
thousands)") +
        xlab("Department") + theme(legend.position="bottom"
                      ,plot.title = element_text(size=10, face="bold")) +
        scale_fill_manual(values= c("#5F9EA0", "#E1B378"))+
        ggtitle("pay and education") + labs(fill =
"Gender")+scale_y_continuous(labels = function(x) format(x, scientific =
FALSE))

    ) })

  output$plotage <- renderPlotly({
    ggplotly(
      ggplot(pg, aes(!!input$variable))+
        geom_density(fill="#69b3a2", color="#e9ecef",
alpha=0.8)+ylab("values in percentage")+scale_y_continuous(labels =
function(x) paste0(x*100, "%"))+
        coord_flip() +#+scale_y_continuous(labels = function(x) format(x,
scientific = FALSE))
      scale_fill_manual(values= c("#5F9EA0", "#E1B378"))



    )})
  output$plotsen <- renderPlotly({
    ggplotly(ggplot() + geom_line(aes(y = BasePay, x = Seniority, colour =
Gender), size=1.5,
                   data = pg, stat="identity")+
          scale_y_continuous(labels = function(x) format(x, scientific =
FALSE))+ggtitle("Seniority level and BasePay")+
            scale_fill_manual(values= c("#5F9EA0", "#E1B378"))+
theme(#legend.position="bottom"
                                        plot.title = element_text(size=8,
face="bold"))
      )})

   output$plotline <- renderPlotly({
      ggplotly(ggplot() + theme_bw() +
         geom_bar(aes(!!input$variable, y = BasePay, fill = Gender), data = pg,
stat="identity")+

            scale_fill_manual(values= c("#5F9EA0",
"#E1B378"))+labs(y="BasePay in thousands") +

            theme(plot.title = element_text(size = 10, face = "bold"),
               axis.text.x=element_text(size=7,
                                 angle = 30,
                                 vjust=.5),  # X axis text
                axis.text.y=element_text(size=7)) + # Y axis text)

            theme(legend.key.size = unit(1, 'cm'))+
            coord_flip()+ggtitle(paste(input$variable, "and Basepay"))
+scale_y_continuous(labels = function(x) format(x, scientific = FALSE))
      ) })



   output$plotbonus <- renderPlotly({
      ggplotly(ggplot() + theme_bw() +
               geom_bar(aes(!!input$variable, y = Bonus, fill = Gender), data =
pg, stat="identity")+
               scale_fill_manual(values= c("#5F9EA0",
"#E1B378"))+labs(y="Bonus in thousands") +
               ggtitle("comapring Bonus") +

            theme(plot.title = element_text(size = 10, face = "bold"),
                    axis.text.x=element_text(size=7,
                                             angle = 30,
                                             vjust=.5),  # X axis text
                     axis.text.y=element_text(size=7)) + # Y axis text)

                theme(legend.key.size = unit(1, 'cm'))+scale_fill_manual(values=
c("#5F9EA0", "#E1B378"))+
                coord_flip()+ggtitle(paste(input$variable, "and Bonus"))
+scale_y_continuous(labels = function(x) format(x, scientific = FALSE))
        ) })


    #  For Data table
    output$pgtable <- DT::renderDataTable(pg, server=FALSE)






}



shinyApp(ui, server)
```

Reference:

Data Source: Glassdoor- Analyze Gender Pay Gap for different job titles based on the gender,Neelima Jauhari “Glassdoor Gender Pay Gap.csv” “https://www.kaggle.com/nilimajauhari/glassdoor-analyze-gender-pay-gap?select=Glassdoor+Gender+Pay+Gap.csv”