

Medicaid Chronic Conditions, Inpatient Admissions and Emergency Room Visits by Zip Code: Beginning 2012

Code ▼

Url: <https://health.data.ny.gov/Health/Medicaid-Chronic-Conditions-Inpatient-Admissions-a/2yck-xisk>
(<https://health.data.ny.gov/Health/Medicaid-Chronic-Conditions-Inpatient-Admissions-a/2yck-xisk>)

Summary:

- Contains information on selected chronic health conditions
- Concerns the Medicaid population at the zip code level
- Posting Frequency: Annually
- Organization: Office of Quality and Patient Safety
- Time Period Beginning 2012 to 2014
- Granularity: Hospital
- Dataset Owner: Bureau of Health Informatics

Notes:

- The data is run on all Medicaid recipients during a 12 month period
- Chronic conditions are identified through use of services and pharmacy
- Medicaid enrollees having a chronic health condition outside of the service period, are not reflected
- Any condition where the number of unique beneficiaries was **20** or less were **suppressed**.

Dimensions and Other Statistics:

- Rows: 98.7K
- Columns: 11

Hide

```
library(corrplot)
```

```
Error in library(corrplot) : there is no package called corrplot
```

Hide

```
mydata = read.csv("./ER-DataSet.csv")  
mydata
```

Y...	Zip.Code	Primary.County	Dual.Eligible	Major.Diagnostic.Category
<int>	<int>	<fctr>	<fctr>	<fctr>
2012	10001	New York	Dual	Diabetes Mellitus
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascu
2012	10001	New York	Dual	Diseases And Disorders Of The Respiratory S
2012	10001	New York	Dual	Mental Diseases And Disorders
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascu
2012	10001	New York	Dual	Mental Diseases And Disorders
2012	10001	New York	Dual	Substance Abuse
2012	10001	New York	Dual	Mental Diseases And Disorders
2012	10001	New York	Dual	Substance Abuse
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascu

1-10 of 98,691 rows | 1-5 of 11 columns Previous 1 2 3 4 5 6 ... 100 Next

County enrollment missing!

Hide

```
renamed_data = rename(mydata,
                        Zip = Zip.Code, PCnty = Primary.County, Dual = Dual.Eligible,
                        MDC = Major.Diagnostic.Category, EDC = Episode.Disease.Categor
y,
                        BC = Beneficiaries.with.Condition,
                        BA = Beneficiaries.with.Admissions,
                        TIA = Total.Inpatient.Admissions,
                        TBERV = Beneficiaries.with.ER.Visits,
                        TERV = Total.ER.Visits)
(renamed_data)
```

Y...	Zip	PCnty	Dual	MDC
<int>	<int>	<fctr>	<fctr>	<fctr>
2012	10001	New York	Dual	Diabetes Mellitus
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascular System
2012	10001	New York	Dual	Diseases And Disorders Of The Respiratory System
2012	10001	New York	Dual	Mental Diseases And Disorders

Y...	Zip	PCnty	Dual	MDC
<int>	<int>	<fctr>	<fctr>	<fctr>
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascular System
2012	10001	New York	Dual	Mental Diseases And Disorders
2012	10001	New York	Dual	Substance Abuse
2012	10001	New York	Dual	Mental Diseases And Disorders
2012	10001	New York	Dual	Substance Abuse
2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascular System
1-10 of 98,691 rows 1-5 of 11 columns				
Previous 1 2 3 4 5 6 ... 100 Next				
< >				

Hide

```
(summary(renamed_data[c("Dual", "BC", "BA", "TIA", "TBERV", "TERV")])) )
```

	Dual	BC	BA	TIA	TBERV
TERV					
Dual	:38324	Min. : 21.0	Min. : 0.00	Min. : 0.0	Min. : 0.0
0	Min. : 0.0				
Non-Dual:60367		1st Qu.: 32.0	1st Qu.: 14.00	1st Qu.: 25.0	1st Qu.: 13.0
0	1st Qu.: 33.0				
		Median : 55.0	Median : 25.00	Median : 51.0	Median : 23.0
0	Median : 69.0				
		Mean : 139.1	Mean : 50.54	Mean : 111.5	Mean : 56.4
1	Mean : 172.3				
		3rd Qu.: 120.0	3rd Qu.: 51.00	3rd Qu.: 113.0	3rd Qu.: 51.0
0	3rd Qu.: 162.0				
		Max. : 7796.0	Max. : 1788.00	Max. : 4099.0	Max. : 3482.0
0	Max. : 8977.0				
< >					

Hide

```

outlier_out_data = filter(renamed_data, !BC%in% boxplot.stats(BC)$out,
                          !BA%in% boxplot.stats(BA)$out,
                          !TIA%in% boxplot.stats(TIA)$out,
                          !TBERV%in% boxplot.stats(TBERV)$out,
                          !TERV%in% boxplot.stats(TERV)$out
                          )
plot_ly(outlier_out_data, x = ~Year, y = ~BC, type = 'box', name = "BC") %>%
  add_trace(y = ~BA, name = "BA") %>%
  add_trace(y = ~TIA, name = "TIA") %>%
  add_trace(y = ~TBERV, name = "TBERV") %>%
  add_trace(y = ~TERV, name = "TERV") %>%
  layout(yaxis = list(title = ''), boxmode = 'group')

```

'layout' objects don't have these attributes: 'boxmode'

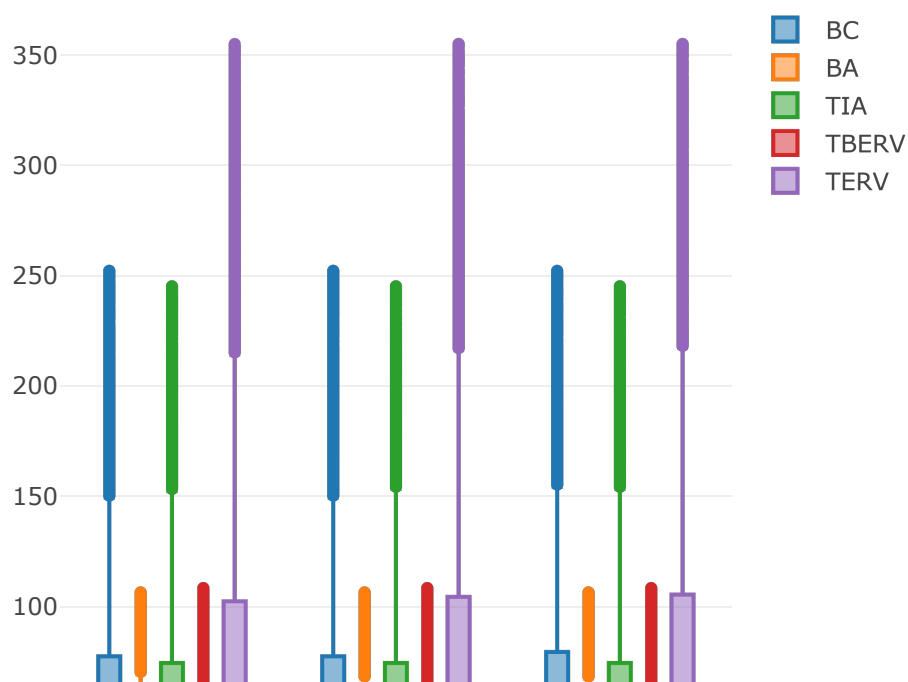
Valid attributes include:

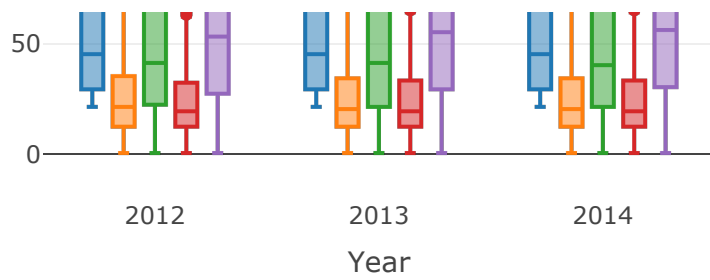
'font', 'title', 'titlefont', 'autosize', 'width', 'height', 'margin', 'paper_bgcolor', 'plot_bgcolor', 'separators', 'hidesources', 'smith', 'showlegend', 'xaxis', 'yaxis', 'ternary', 'scene', 'geo', 'mapbox', 'radialaxis', 'angularaxis', 'direction', 'orientation', 'dragmode', 'hovermode', 'hoverlabel', 'legend', 'annotations', 'shapes', 'images', 'updatemenus', 'sliders', 'calendar', 'barmode', 'bargap', 'mapType'

'layout' objects don't have these attributes: 'boxmode'

Valid attributes include:

'font', 'title', 'titlefont', 'autosize', 'width', 'height', 'margin', 'paper_bgcolor', 'plot_bgcolor', 'separators', 'hidesources', 'smith', 'showlegend', 'xaxis', 'yaxis', 'ternary', 'scene', 'geo', 'mapbox', 'radialaxis', 'angularaxis', 'direction', 'orientation', 'dragmode', 'hovermode', 'hoverlabel', 'legend', 'annotations', 'shapes', 'images', 'updatemenus', 'sliders', 'calendar', 'barmode', 'bargap', 'mapType'





Hide

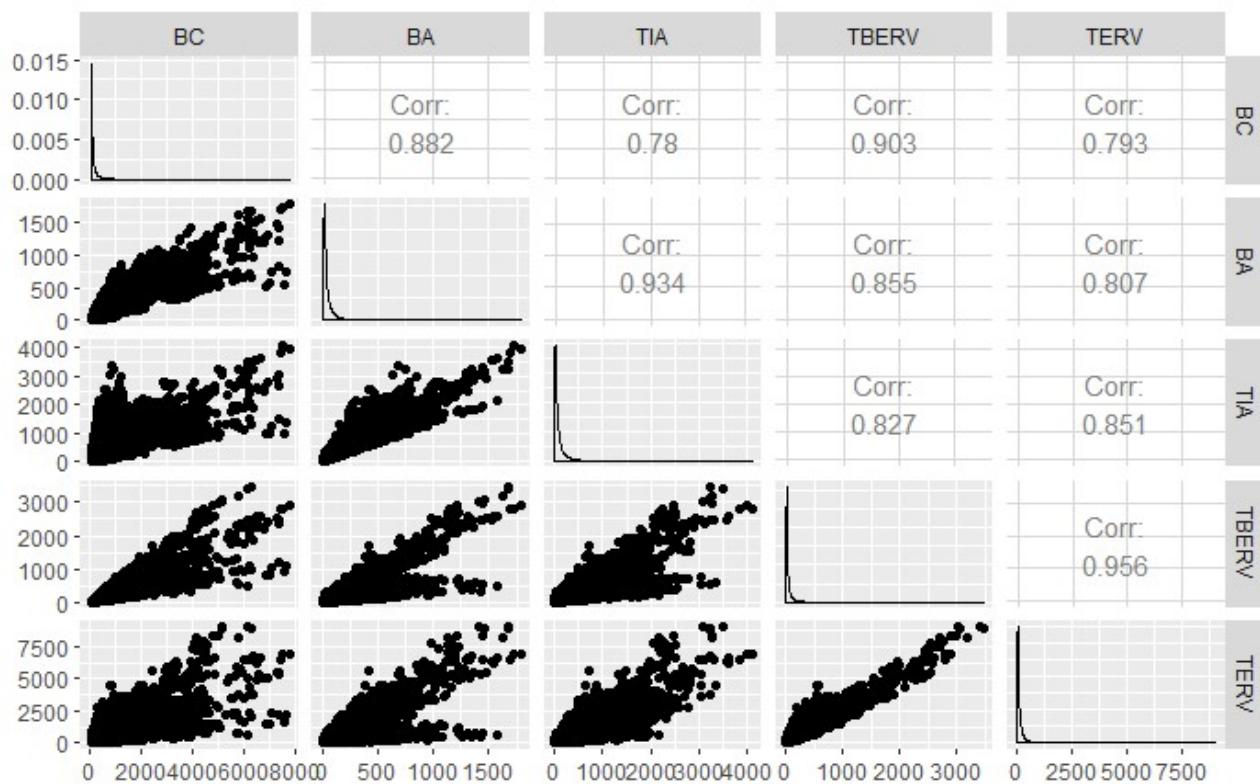
```
columns = data.frame(renamed_data[ , !names(renamed_data) %in% c("Year", "MDC", "ED  
C", "Zip", "PCnty", "Dual") ] )  
ggpairs(columns )
```

```
plot: [1,1] [==-----]
-----] 4% est: 0s
plot: [1,2] [=====-----]
-----] 8% est: 9s
plot: [1,3] [=====-----]
-----] 12% est: 7s
plot: [1,4] [=====-----]
-----] 16% est: 6s
plot: [1,5] [=====-----]
-----] 20% est: 5s
plot: [2,1] [=====-----]
-----] 24% est: 4s
plot: [2,2] [=====-----]
-----] 28% est: 6s
plot: [2,3] [=====-----]
-----] 32% est: 5s
plot: [2,4] [=====-----]
-----] 36% est: 5s
plot: [2,5] [=====-----]
-----] 40% est: 4s
plot: [3,1] [=====-----]
-----] 44% est: 4s
plot: [3,2] [=====-----]
-----] 48% est: 4s
plot: [3,3] [=====-----]
-----] 52% est: 4s
plot: [3,4] [=====-----]
-----] 56% est: 3s
plot: [3,5] [=====-----]
-----] 60% est: 3s
plot: [4,1] [=====-----]
-----] 64% est: 2s
plot: [4,2] [=====-----]
-----] 68% est: 2s
plot: [4,3] [=====-----]
-----] 72% est: 2s
plot: [4,4] [=====-----]
-----] 76% est: 2s
plot: [4,5] [=====-----]
-----] 80% est: 2s
plot: [5,1] [=====-----]
-----] 84% est: 1s
plot: [5,2] [=====-----]
-----] 88% est: 1s
plot: [5,3] [=====]
=====] 92% est: 1s
plot: [5,4] [=====]
```

```
=====] 96% est: 0s
```

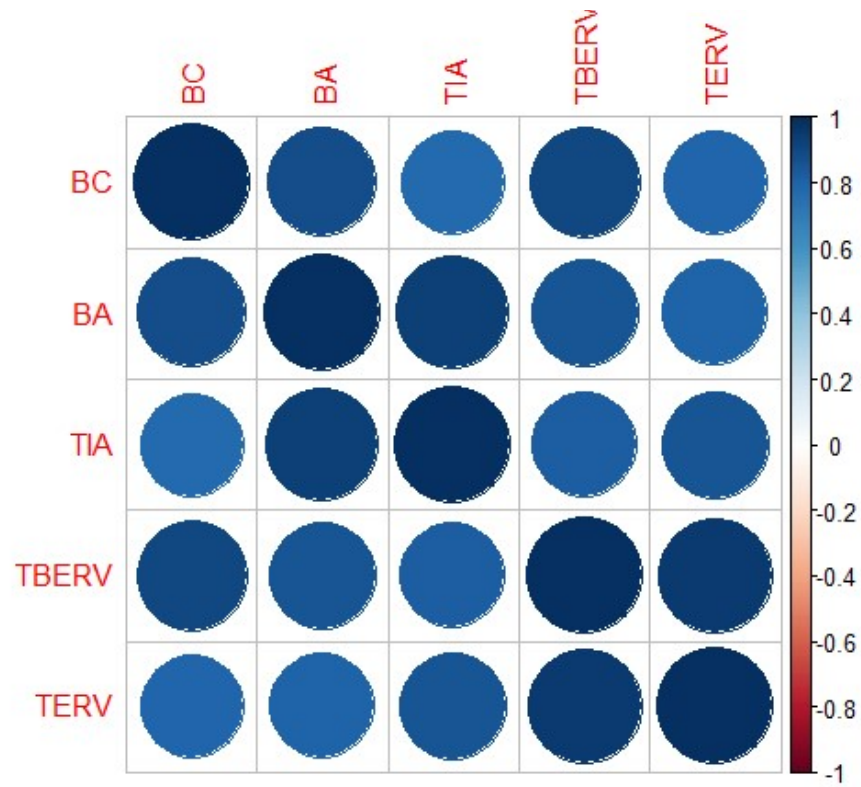
```
plot: [5,5] [=====]
```

```
=====]100% est: 0s
```



Hide

```
corrplot( cor(renamed_data[c("BC", "BA", "TIA", "TBERV", "TERV")]) )
```



Hide

```
d_stan = as.data.frame(scale(renamed_data[c("BC", "BA", "TIA", "TBERV", "TERV")]))
res1b = factanal(d_stan, factors = 2, rotation = "none", na.action = na.omit)
res1b$loadings
```

```
Loadings:
      Factor1 Factor2
BC      0.925
BA      0.960 -0.270
TIA     0.915 -0.200
TBERV   0.963  0.260
TERV    0.918  0.273

      Factor1 Factor2
SS loadings    4.385  0.255
Proportion Var  0.877  0.051
Cumulative Var  0.877  0.928
```

Hide

```
summary(renamed_data[5])
```


	MDC
Diabetes Mellitus	:11416
Diseases And Disorders Of The Cardiovascular System	:30294
Diseases And Disorders Of The Respiratory System	:11329
HIV Infection	: 1077
Mental Diseases And Disorders	:32481
Newborns And Other Neonates	: 86
Substance Abuse	:12008

Hide

```
ax_data = renamed_data
levels(ax_data$MDC) <- c("Diabetes", "Cardiovascular", "Respiratory ",
                        "HIV", "Mental", "Newborns", "Subtnc-Abuse")

plot_ly(ax_data, x = ~MDC) %>%
  layout(title = "Frequency of Each Categor",
         yaxis = list(title = ''), xaxis = list(title = "", tickangle = 45),
         margin = list(b = 250))
```

No trace type specified:

Based on info supplied, a 'histogram' trace seems appropriate.

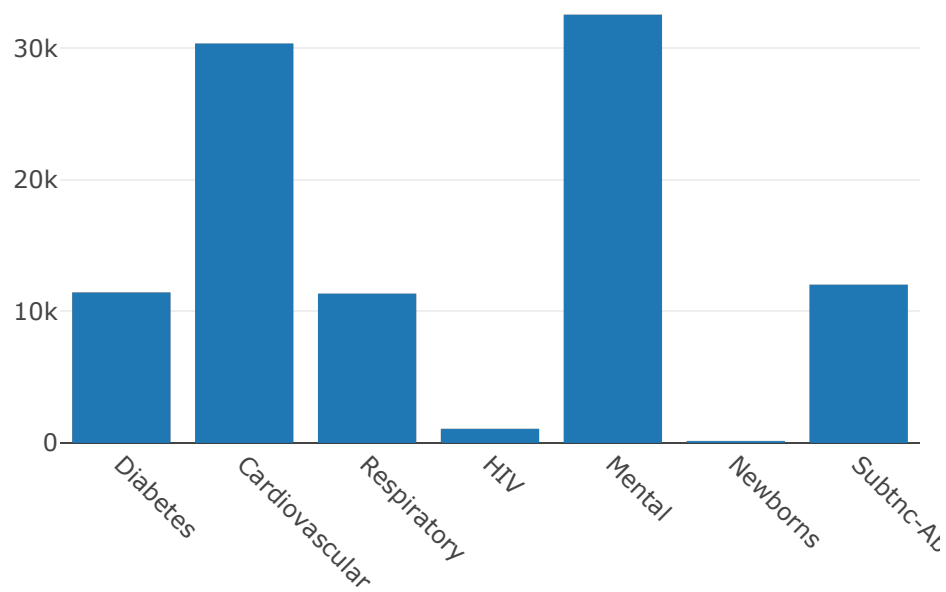
Read more about this trace type -> <https://plot.ly/r/reference/#histogram>

No trace type specified:

Based on info supplied, a 'histogram' trace seems appropriate.

Read more about this trace type -> <https://plot.ly/r/reference/#histogram>

Frequency of Each Categor



Hide

```
plot_ly(renamed_data, x = ~MDC) %>%  
  layout(title = "Frequency of Each Categor",  
    yaxis = list(title = ''), xaxis = list(title = "", tickangle = 45),  
    margin = list(b = 250))
```

No trace type specified:

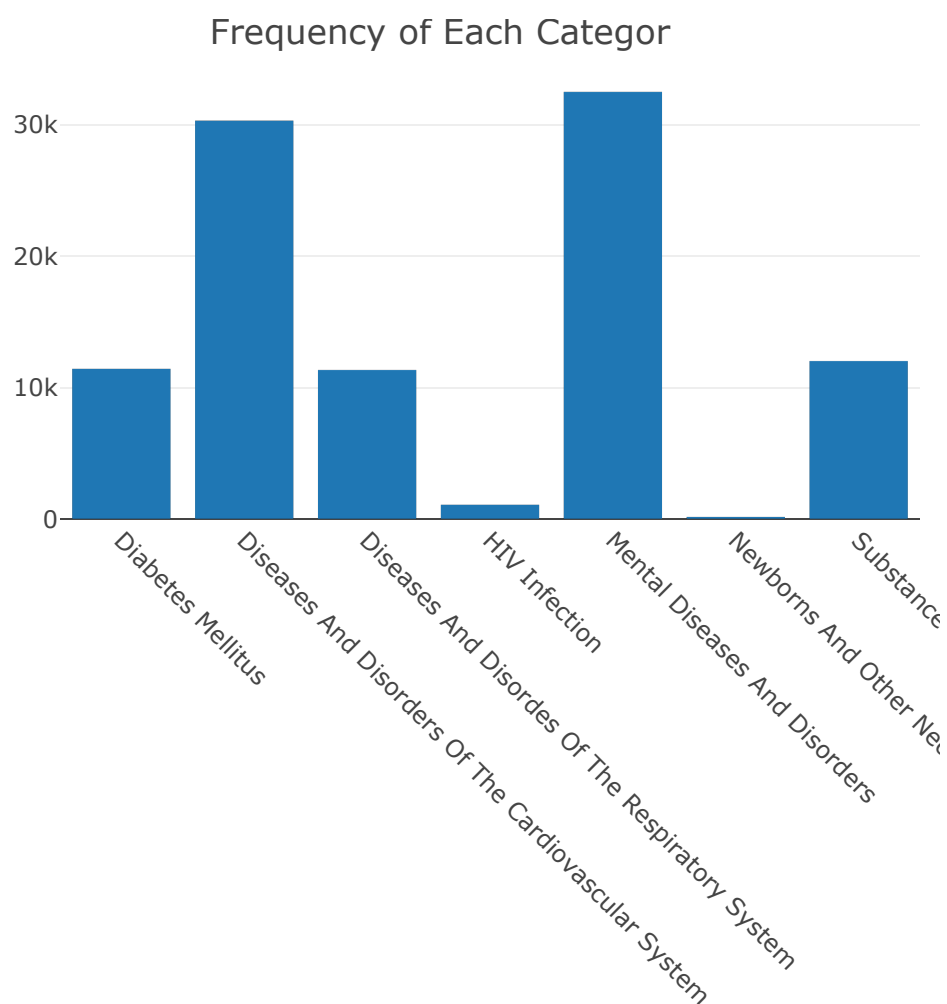
Based on info supplied, a 'histogram' trace seems appropriate.

Read more about this trace type -> <https://plot.ly/r/reference/#histogram>

No trace type specified:

Based on info supplied, a 'histogram' trace seems appropriate.

Read more about this trace type -> <https://plot.ly/r/reference/#histogram>



Hide

```

MDC_Dual = renamed_data[, c("MDC", "Dual")] %>%
  group_by(MDC, Dual) %>% summarise(n())
colnames(MDC_Dual) = c("MDC", "Dual", "Frequency")
levels(MDC_Dual$MDC) <- c("Diabetes", "Cardiovascular", "Respiratory ",
                           "HIV", "Mental", "Newborns", "Subtnc-Abuse")

plot_ly(MDC_Dual, x = ~MDC, y = ~Frequency, color = ~Dual) %>%
  layout(title = "Frequency vs Dual Eligiblity",
         yaxis = list(title = ''), xaxis = list(title = "", tickangle = 45),
         margin = list(b = 250))

```

No trace type specified:

Based on info supplied, a 'bar' trace seems appropriate.

Read more about this trace type -> <https://plot.ly/r/reference/#bar>

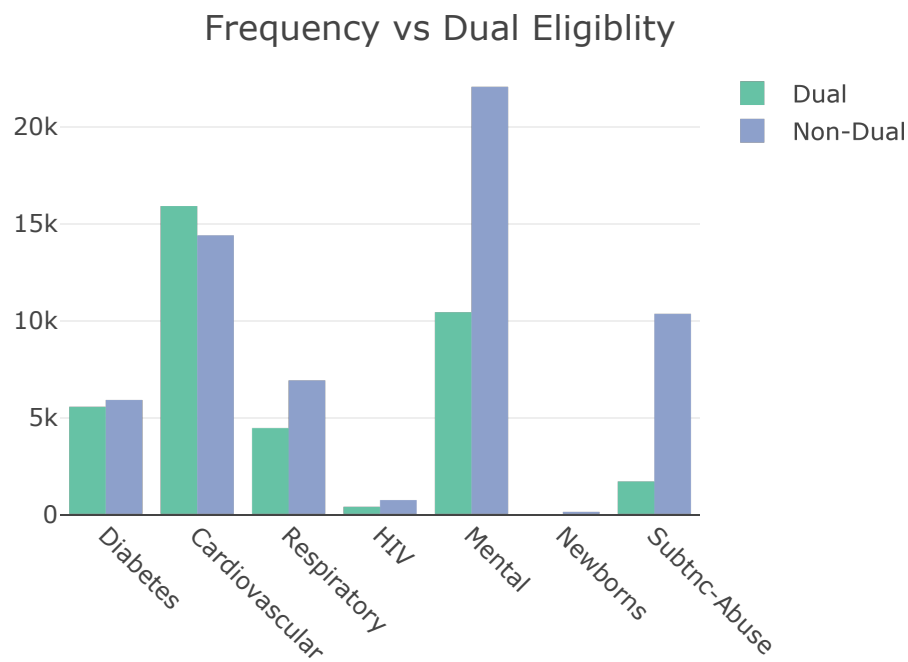
minimal value for n is 3, returning requested palette with 3 different levels

No trace type specified:

Based on info supplied, a 'bar' trace seems appropriate.

Read more about this trace type -> <https://plot.ly/r/reference/#bar>

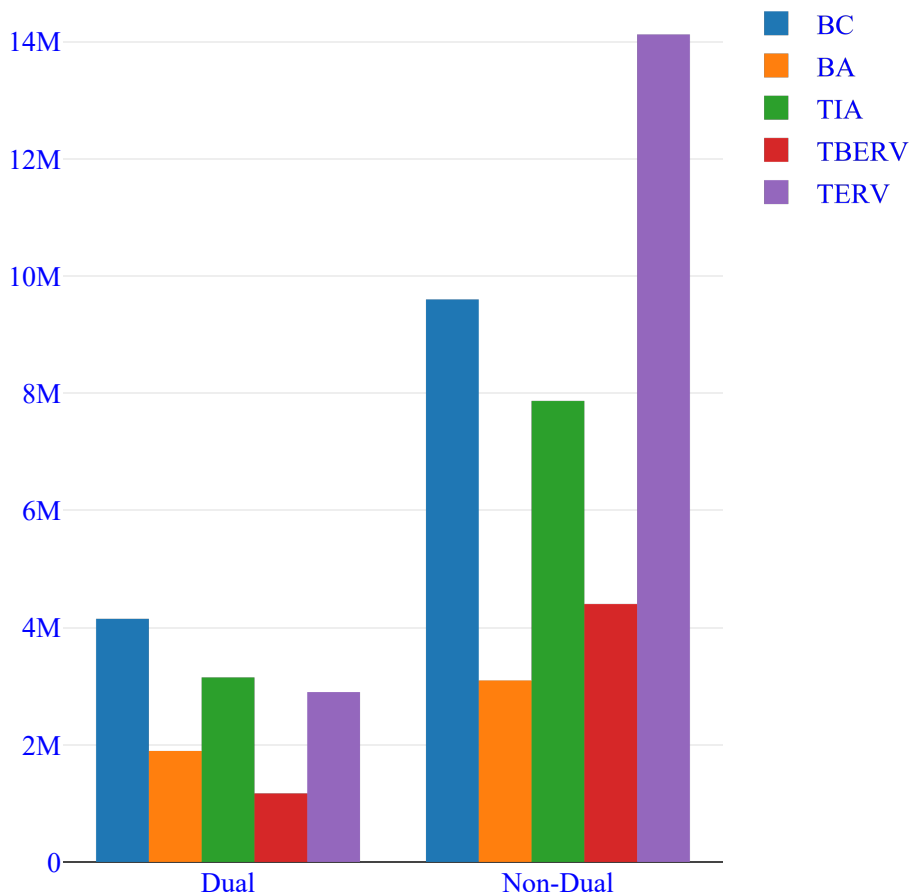
minimal value for n is 3, returning requested palette with 3 different levels



[Hide](#)

```
by_Dual_data = renamed_data[, c("Dual", "BC", "BA", "TIA", "TBERV", "TERV")] %>%
  group_by(Dual) %>%
    summarise( BC = sum(BC), BA = sum(BA),
               TIA = sum(TIA),
               TBERV = sum(TBERV),
               TERV = sum(TERV))
t <- list( family = "sans serif", size = 14, color = 'blue')
plot_ly(by_Dual_data, x = ~Dual, y = ~BC, type = 'bar', name = "BC") %>%
  add_trace(y = ~BA, name = "BA") %>%
  add_trace(y = ~TIA, name = "TIA") %>%
  add_trace(y = ~TBERV, name = "TBERV") %>%
  add_trace(y = ~TERV, name = "TERV") %>%
  layout( title = "Population Count",
          font = t,
          yaxis = list(title = ''), xaxis = list(title = ""), barmode = 'group')
```

Population Count

[Hide](#)

```

ui <- fluidPage(
  selectInput("categ", "Name of Category",
    c("Diabetes Mellitus" ,
      "Diseases And Disorders Of The Cardiovascular System",
      "Diseases And Disorders Of The Respiratory System",
      "HIV Infection",
      "Mental Diseases And Disorders",
      "Newborns And Other Neonates",
      "Substance Abuse"
    ))
  , # Now outputs
  plotlyOutput("my_plot_name")

)
server <- function(input, output) {

  output$my_plot_name <-

    renderPlotly({

      MDC_EDC = renamed_data[, c("MDC", "EDC")] %>% filter(MDC == input$categ )
      MDC_EDC <- lapply(MDC_EDC, factor)
      EDC_factor = as.factor( unlist(MDC_EDC[2]) )
      df_EDC = data.frame(table(EDC_factor))
      names(df_EDC) <- c("EDC_Category", "Freq")

      plot_ly(df_EDC, x = ~EDC_Category, y = ~Freq, type = 'bar', insidetextfont = list
(color = '#FFFFFF'), hoverinfo = 'text') %>%
      layout( title = paste("Category: ", input$categ),
        xaxis = list(title = "", tickangle = 45), yaxis = list(title = ""),
        margin = list(b = 200),
        font = t )
    })

}
shinyApp(server = server, ui = ui)

```

Listening on http://127.0.0.1:4860

Ignoring explicitly provided widget ID "10e426fd50fa"; Shiny doesn't use them
Ignoring explicitly provided widget ID "10e428ee4343"; Shiny doesn't use them

Hide

```

runApp(list(
  ui = basicPage(
    #h2('The attributes to select'),
    checkboxGroupInput("columns","Select Columns",
                      choices = c("BC", "BA", "TIA", "TBERV", "TERV"), inline = T),
    plotlyOutput("my_plot_name")
  ),
  server = function(input, output) {
    output$my_plot_name <- renderPlotly({
      if( length(input$columns) == 0 ){
        plot_ly() %>% layout()
        #dfzero <- by_MDC_data[,c("MDC", "BC")]
        #names(dfzero) <- c("MDC", "BC")
        #plot_ly(dfzero, x = ~MDC, y = ~BC, type = 'bar', name = "TERV") %>%
        # layout(title = "Total count of each Categor",
        #        yaxis = list(title = ''), xaxis = list(title = ""), barmode = 'group')
      }
      #if(length(input$columns) == 1){
      # cols <- c("MDC", input$columns)
      # df <- data.frame(by_MDC_data[,cols])
      # names(df) <- c("MDC", "input_col")
      # plot_ly(df, x = ~MDC, y = ~input_col, type = 'bar', name = "TERV") %>%
      # layout(title = "Total count of each Categor",
      #        yaxis = list(title = ''), xaxis = list(title = "", tickangle = -90),
      #        margin = list(b = 200), barmode = 'group')
      #}
      else{
        cols <- c("MDC", input$columns)
        df <- data.frame(by_MDC_data)
        names(df) <- c("MDC", "BC", "BA", "TIA", "TBERV", "TERV")
        df$MDC <- factor(df$MDC, levels = df[["MDC"]])
        p = plot_ly(df, x = ~MDC, type = 'bar', name = "BC") %>%
          layout( title = "Total count of each Categor",
                  yaxis = list(title = ''), xaxis = list(title = "", tickangle = 45),
                  margin = list(b = 200),
                  barmode = 'group')
        if ("BC" %in% cols){ p = add_trace(p, y = ~BC, name = "BC")}
        if ("BA" %in% cols){ p = add_trace(p, y = ~BA, name = "BA")}
        if ( "TIA" %in% cols){ p = add_trace(p, y = ~TIA, name = "TIA") }
        if ( "TBERV" %in% cols){ p = add_trace(p, y = ~TBERV, name = "TBERV") }
        if ( "TERV" %in% cols){ p = add_trace(p, y = ~TERV, name = "TERV") }
        p
      }
    })
  }
))

```

```

Listening on http://127.0.0.1:4860
Ignoring explicitly provided widget ID "10e42252220f"; Shiny doesn't use them
No trace type specified and no positional attributes specified
No trace type specified:
  Based on info supplied, a 'scatter' trace seems appropriate.
  Read more about this trace type -> https://plot.ly/r/reference/#scatter
No scatter mode specified:
  Setting the mode to markers
  Read more about this attribute -> https://plot.ly/r/reference/#scatter-mode
Error in data.frame: object 'by_MDC_data' not found
Stack trace (innermost first):
  80: data.frame
  79: "plotly"::"ggplotly" [#39]
  78: func
  77: origRenderFunc
  76: output$my_plot_name
  1: runApp
Error in data.frame: object 'by_MDC_data' not found
Stack trace (innermost first):
  80: data.frame
  79: "plotly"::"ggplotly" [#39]
  78: func
  77: origRenderFunc
  76: output$my_plot_name
  1: runApp

```

Hide

```

detailed_data = read.csv("NewData_Detailed.csv")
detailed_data

```

Y...	Zip.Code	County	Total.Beneficiaries	Dual.Eligible.Beneficiaries
<int>	<int>	<fctr>	<int>	<int>
2012	6390	Suffolk	4	1
2012	10001	New York	10704	2478
2012	10002	New York	58639	10748
2012	10003	New York	4668	1163
2012	10004	New York	635	268
2012	10005	New York	1052	66
2012	10006	New York	100	14
2012	10007	New York	620	264
2012	10008	New York	86	19

Y...	Zip.Code	County	Total.Beneficiaries	Dual.Eligible.Beneficiaries						
<int>	<int>	<fctr>	<int>	<int>						
2012	10009	New York	18105	4246						
1-10 of 6,335 rows 1-6 of 20 columns			Previous	1	2	3	4	5	6	... 100 Next

Hide

```
selected_columns = detailed_data[, c("Year", "Zip.Code", "County", "Total.Beneficiaries")] %>%
  rename(Zip = Zip.Code, PCnty = County, TB = Total.Beneficiaries)
joined_data = inner_join(renamed_data, selected_columns)
```

Joining, by = c("Year", "Zip", "PCnty")
Column `PCnty` joining factors with different levels, coercing to character vector

Hide

```
cnty_poplulation = detailed_data[, c("County", "Total.Beneficiaries")] %>%
  group_by(County) %>% summarise(TBC = sum(Total.Beneficiaries)) %>%
  rename(PCnty = County)
joined_data = inner_join(joined_data, cnty_poplulation)
```

Joining, by = "PCnty"
Column `PCnty` joining character vector and factor, coercing into character vector

Hide

```
head(joined_data)
```

Y...	Zip	PCnty	D...	MDC
<int>	<int>	<chr>	<fctr>	<fctr>
1 2012	10001	New York	Dual	Diabetes Mellitus
2 2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascular System
3 2012	10001	New York	Dual	Diseases And Disorders Of The Respiratory System
4 2012	10001	New York	Dual	Mental Diseases And Disorders
5 2012	10001	New York	Dual	Diseases And Disorders Of The Cardiovascular System
6 2012	10001	New York	Dual	Mental Diseases And Disorders
6 rows 1-6 of 13 columns				

Hide

```
by_cnty_data = renamed_data[, c("PCnty", "BC", "BA", "TIA", "TBERV", "TERV")]%>%
  group_by(PCnty) %>%
    summarise( BC = sum(BC), BA = sum(BA),
               TIA = sum(TIA),
               TBERV = sum(TBERV),
               TERV = sum(TERV))
(by_cnty_data)
```

PCnty <fctr>	BC <int>	BA <int>	TIA <int>	TBERV <int>	TERV <int>					
Albany	143457	51089	105291	77272	304956					
Allegany	17788	4721	8118	9031	24357					
Bronx	2130026	869491	2143001	919118	2598622					
Broome	110526	33651	59392	53265	180718					
Cattaraugus	42618	13111	21710	21529	67785					
Cayuga	37153	12447	24549	19111	65360					
Chautauqua	86037	25481	45478	42490	127341					
Chemung	63117	22355	41323	35318	127719					
Chenango	19006	4895	7761	9337	28439					
Clinton	49191	15595	26290	25302	77819					
1-10 of 61 rows		Previous	1	2	3	4	5	6	7	Next

Combine Chroplothe and ShinyApp

Hide

```

ui <- fluidPage(
  radioButtons("attr", "Name of Attribute", c("BC","BA", "TIA" , "TBERV" , "TERV"), i
nline = TRUE), # Now outputs
  leafletOutput("mymap")
)
server <- function(input, output) {

  output$mymap <- renderLeaflet({

    adjusted_data <- by_cnty_data[,c("PCnty", input$attr)]
    names(adjusted_data) <- c("NAME_2", "col_name")

    # get county level spatial polygons for the United States
    counties <- getData("GADM", country = "USA", level = 2)

    # filter down to just New York State Counties
    counties <- counties[counties@data$NAME_1 == "New York",]
    bins <- c(0, 10, 20, 50, 100, 200, 500, 1000, Inf)
    pal <- colorBin("YlOrRd", domain = density, bins = bins)

    ## In our data we have St Lawrence but in our SP obkect we have Saint lawrence, s
o we
    ## fix it by gsub()
    adjusted_data$NAME_2 = gsub("St Lawrence", "Saint Lawrence", adjusted_data$NAME_2)
    counties@data = left_join(counties@data, adjusted_data)

    state_popup <- paste0("<strong>County: </strong>",
                          counties$NAME_2,
                          "<br><strong>Attribute is : </strong>", input$attr,
                          "<br><strong> Value : </strong>", counties$col_name/100)

    counties %>% leaflet() %>% addTiles() %>%
      addPolygons(
        fillColor = ~pal(col_name/100),
        weight = 2,
        opacity = 1,
        color = "blue", # we can change it or remove it
        dashArray = "3",
        fillOpacity = 0.7,
        highlight = highlightOptions(
          weight = 5,
          color = "#666",
          dashArray = "",

```

```

        fillOpacity = 0.7,
        bringToFront = TRUE),
    popup = state_popup
  ) %>%

  addLegend("bottomright", pal = pal, values = ~col_name/100,
    title = ,
    #labFormat = labelFormat(prefix = "$"),
    opacity = 1
  )

})

}
shinyApp(server = server, ui = ui)

```

```

Listening on http://127.0.0.1:4860
trying URL 'http://biogeo.ucdavis.edu/data/gadm2.8/rds/USA_adm2.rds'
Content type 'application/octet-stream' length 13943951 bytes (13.3 MB)
downloaded 13.3 MB

Joining, by = "NAME_2"
Joining, by = "NAME_2"
Joining, by = "NAME_2"

```

Now we map MDC

Hide

```

# library(dplyr)
# MCD_cnty = renamed_data %>% filter(MDC == "Diabetes Mellitus")
# MCD_cnty = MCD_cnty[, c("PCnty", "BC", "BA", "TIA", "TBERV", "TERV")] %>%
#   group_by(PCnty) %>% summarise(BC = sum(BC), BA = sum(BA),
#   TIA = sum(TIA), TBERV = sum(TBERV), TERV = sum(TERV))
# MCD_cnty

```

Hide

```

ui <- fluidPage(
  selectInput("attr", "Name of MDC Category",
    c("Diabetes Mellitus",
      "Diseases And Disorders Of The Cardiovascular System",
      "Diseases And Disorders Of The Respiratory System" ,
      "HIV Infection" ,
      "Mental Diseases And Disorders",
      "Newborns And Other Neonates",
      "Substance Abuse")
  ), # Now outputs
  selectInput("var", "Name of Attribute", c("BC","BA", "TIA" , "TBERV" , "TERV")),
  leafletOutput("mymap")
)
server <- function(input, output) {

  output$mymap <- renderLeaflet({

    MCD_cnty = joined_data %>% filter(MDC == input$attr)
    MCD_cnty = MCD_cnty %>% mutate( BC = BC*10^3/TBC, BA = BA*10^3/TBC, TIA = TIA*10^
3/TBC,
                                TBERV = TBERV*10^3/TBC, TERV = TERV*10^3/TBC)
    MCD_cnty = MCD_cnty[, c("PCnty", "BC","BA", "TIA" , "TBERV" , "TERV")] %>%
      group_by(PCnty) %>% summarise(BC = sum(BC), BA = sum(BA),
                                TIA = sum(TIA), TBERV = sum(TBERV), TERV = sum(TERV)
                                )

    ## To keep it dafe:
    #MCD_cnty = renamed_data %>% filter(MDC == input$attr)
    #MCD_cnty = MCD_cnty[, c("PCnty", "BC","BA", "TIA" , "TBERV" , "TERV")] %>%
    #  group_by(PCnty) %>% summarise(BC = sum(BC), BC = sum(BC), BA = sum(BA),
    #                                TIA = sum(TIA), TBERV = sum(TBERV), TERV = sum(TERV))

    adjusted_data <- MCD_cnty[,c("PCnty", input$var)]
    names(adjusted_data) <- c("NAME_2", "col_name")

    # get county level spatial polygons for the United States
    counties <- getData("GADM", country = "USA", level = 2)

    # filter down to just New York State Counties
    counties <- counties[counties@data$NAME_1 == "New York",]
    bins <- c(0, 25, 45, 60, 80, 170, 250, 400, 700, Inf)
    pal <- colorBin("YlOrRd", domain = density, bins = bins)

    ## In our data we have St Lawrence but in our SP obkect we have Saint lawrence, s
o we
    ## fix it by gsub()
    adjusted_data$NAME_2 = gsub("St Lawrence", "Saint Lawrence", adjusted_data$NAME_2)
  })
}

```

```

counties@data = left_join(counties@data, adjusted_data)

#pal <- brewer.pal(15, "YlGnBu")

state_popup <- paste0("<strong>County: </strong>",
                      counties$NAME_2,
                      "<br><strong>MDC category : </strong>", input$attr,
                      "<br><strong> Value per 1K : </strong>", round(counties$col_
name, 3) )

counties %>% leaflet() %>% addTiles() %>%
  addPolygons(
    fillColor = ~pal(col_name),
    weight = 2,
    opacity = 1,
    color = "blue", # we can change it or remove it
    dashArray = "3",
    fillOpacity = 0.7,
    highlight = highlightOptions(
      weight = 5,
      color = "#666",
      dashArray = "",
      fillOpacity = 0.7,
      bringToFront = TRUE),
    popup = state_popup
  ) %>%

  addLegend("bottomright", pal = pal, values = ~col_name,
    title = ,
    #labFormat = labelFormat(prefix = "$"),
    opacity = 1
  )

})

}
shinyApp(server = server, ui = ui)

```

```

Listening on http://127.0.0.1:4860
Joining, by = "NAME_2"
Joining, by = "NAME_2"
Joining, by = "NAME_2"

```