EJPracticoentiendetutabladedatos.R

apalmad

2021-01-22

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# 1.1 INSTALAR PAQUETES DE FUNCIONES  
#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# Lista de paquetes de funciones a instalar  
.packages = c("ggplot2", "plotly", "xlsx","scales","stringr","readr","dplyr","psych","readxl","ggpubr")  
  
  
# Instala los paquetes sinó los tienes instalados  
.inst <- .packages %in% installed.packages()  
if(length(.packages[!.inst]) > 0) install.packages(.packages[!.inst])  
  
  
#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# 1.2 CARGAR PAQUETES O CREAR FUNCIONES  
#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
# Carga los paquetes sinó los tienes cargados  
lapply(.packages, require, character.only=TRUE)

## Loading required package: ggplot2

## Loading required package: plotly

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

## Loading required package: xlsx

## Error: package or namespace load failed for 'xlsx':  
## .onLoad failed in loadNamespace() for 'rJava', details:  
## call: fun(libname, pkgname)  
## error: JAVA\_HOME cannot be determined from the Registry

## Loading required package: scales

## Loading required package: stringr

## Loading required package: readr

##   
## Attaching package: 'readr'

## The following object is masked from 'package:scales':  
##   
## col\_factor

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

## Loading required package: psych

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:scales':  
##   
## alpha, rescale

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

## Loading required package: readxl

## Loading required package: ggpubr

## [[1]]  
## [1] TRUE  
##   
## [[2]]  
## [1] TRUE  
##   
## [[3]]  
## [1] FALSE  
##   
## [[4]]  
## [1] TRUE  
##   
## [[5]]  
## [1] TRUE  
##   
## [[6]]  
## [1] TRUE  
##   
## [[7]]  
## [1] TRUE  
##   
## [[8]]  
## [1] TRUE  
##   
## [[9]]  
## [1] TRUE  
##   
## [[10]]  
## [1] TRUE

#####################################  
#  
#Setear el working directory  
#  
# Leeo la tabla de datos  
####################################  
setwd("C:/Users/apalmad/Desktop/Analiza tus Datos/Proyectos R/Caso Practico/Entiende tu Tabla de Datos/Ejercicio 2")  
  
Datos <- read\_xlsx("Datos.xlsx")  
  
#################################  
# Describo las variables  
###############################  
str(Datos)

## tibble [193 x 9] (S3: tbl\_df/tbl/data.frame)  
## $ pais : chr [1:193] "Afghanistan" "Albania" "Algeria" "Angola" ...  
## $ region : chr [1:193] "Asia" "Europe" "Africa" "Africa" ...  
## $ group : chr [1:193] "other" "other" "africa" "africa" ...  
## $ fertility : num [1:193] 5.97 1.52 2.14 5.13 2.17 ...  
## $ ppgdp : num [1:193] 499 3677 4473 4322 9162 ...  
## $ lifeExpF : num [1:193] 49.5 80.4 75 53.2 79.9 ...  
## $ pctUrban : num [1:193] 23 53 67 59 93 64 47 89 68 52 ...  
## $ infantMortality: num [1:193] 124.5 16.6 21.5 96.2 12.3 ...  
## $ LogNepPib : num [1:193] 6.21 8.21 8.41 8.37 9.12 ...

##  
# Se definen funciones  
##  
descriptiva <- function(datos,columnas,colagruparpor) {  
   
 filtrar\_por <- unique(Datos[,colagruparpor])  
 listaout <- list()  
   
 for (g in (1:nrow(filtrar\_por)))   
 {  
 df <- filter(Datos, (Datos[,colagruparpor] == str\_trim(filtrar\_por[g,1])) ) %>%  
 select(c(colagruparpor,columnas))  
 for (i in (2:ncol(df)))  
 {  
 lista <- list(   
 Grupo = str\_trim(filtrar\_por[g,1]),  
 media = apply(df[,i],2,mean),  
 cutmedia = apply(df[,i],2,mean,trim=0.05),  
 StdErr = apply(df[,i],2,sd) / sqrt(apply(df[,i],2,length)),  
 IC95LimInf = apply(df[,i],2,mean) - (1.96 \* apply(df[,i],2,sd) / sqrt(apply(df[,i],2,length))),  
 IC95LimSup = apply(df[,i],2,mean) + (1.96 \* apply(df[,i],2,sd) / sqrt(apply(df[,i],2,length))),  
 mediana= apply(df[,i],2,median),  
 sd= apply(df[,i],2,sd),  
 var = apply(df[,i],2,var),  
 obs = apply(df[,i],2,length),  
 Min = apply(df[,i],2,min),  
 Max= apply(df[,i],2,max),  
 Rango = apply(df[,i],2,max) - apply(df[,i],2,min),  
 RangoIQR = apply(df[,i],2,quantile,prob=0.25),  
 Q1= apply(df[,i],2,quantile,prob=0.75),  
 Q3= apply(df[,i],2,quantile,prob=0.75) - apply(df[,i],2,quantile,prob=0.25),  
 skew= apply(df[,i],2,skew),  
 kurtosis = apply(df[,i],2, kurtosi)  
   
 )  
 listaout[[str\_trim(filtrar\_por[g,1])]] <- lista  
   
 }  
 }  
   
 #Colocando Nombres a las Columnas y Filas a la matriz estadistica descriptiva  
 mt <- matrix(ncol = nrow(filtrar\_por), nrow = 18 )  
 colnames(mt) <- names(listaout)  
 rownames(mt) <- names(listaout[[1]])  
   
 for (i in names(listaout))  
 {  
 mt[,i] <- unlist(listaout[[i]])  
 }   
 mt  
}  
crea\_errdiagram <- function(Datos,columna,agrupapor,titulo="Titulo",xlabel="xlabel",ylabel="ylabel"){  
   
   
 dfsumarize <- Datos %>%  
 group\_by(get(agrupapor)) %>%  
 summarise(  
 sd = sd(get(columna),na.rm = TRUE),  
 media = mean(get(columna)))  
   
 #####  
 # Creo grafico de errores o de medias  
 ####  
 ggplot(dfsumarize, aes(x=reorder(`get(agrupapor)`,media), y=media)) +  
 geom\_line(aes(group = 1)) +  
 geom\_errorbar( aes(ymin=media-sd,ymax=media + sd),width = 0.2) +  
 geom\_point(size = 2) +  
 labs(title = titulo,  
 subtitle = "",  
 caption = "Data source: Analiza tus Datos",  
 x = xlabel,  
 y = ylabel,  
 tag = "Primeros Programas con R") +  
 theme(plot.title = element\_text(size =12, face = "bold", hjust = 0.5 ),  
 plot.subtitle = element\_text(size = 9, hjust = 0.5),  
 plot.caption = element\_text(size = 7),  
 plot.tag = element\_text(size = 7),  
 axis.text = element\_text(size=8),  
 axis.title = element\_text(size=8,face="bold"))  
   
}  
creaboxplot <- function(Datos, var\_x,var\_y,Titulo,xlabel,ylabel="NA"){  
 ggplot(data = Datos) +  
 geom\_boxplot(mapping = aes(  
 x = reorder(get(var\_x),get(var\_y),FUN = median),  
 y = get(var\_y)  
 )) +  
 labs(title = Titulo,  
 subtitle = "",  
 caption = "Data source: Analiza tus Datos",  
 x = xlabel,  
 y = ylabel,  
 tag = "Primeros Programas con R") +  
 theme(plot.title = element\_text(size =12, face = "bold", hjust = 0.5 ),  
 plot.subtitle = element\_text(size = 9, hjust = 0.5),  
 plot.caption = element\_text(size = 7),  
 plot.tag = element\_text(size = 7),  
 axis.text = element\_text(size=8),  
 axis.title = element\_text(size=8,face="bold"))  
   
   
}  
creascatplot <- function(Datos, var\_x,var\_y,Titulo,xlabel,ylabel="NA",categoria="group"){  
 ggplot(data = Datos) +  
 geom\_point(mapping = aes(x = get(var\_x), y = get(var\_y),color=get(categoria))) +  
   
 labs(title = Titulo,  
 subtitle = "",  
 caption = "Data source: Analiza tus Datos",  
 x = xlabel,  
 y = ylabel,  
 tag = "Primeros Programas con R") +  
 theme(plot.title = element\_text(size =12, face = "bold", hjust = 0.5 ),  
 plot.subtitle = element\_text(size = 9, hjust = 0.5),  
 plot.caption = element\_text(size = 7),  
 plot.tag = element\_text(size = 7),  
 axis.text = element\_text(size=8),  
 axis.title = element\_text(size=8,face="bold")) +  
 geom\_smooth(mapping = aes(x = get(var\_x), y = get(var\_y)), orientation = "y")  
 # facet\_grid(. ~ get(categoria))   
   
   
}  
creagrafcorr <- function(Datos,colx,coly,title,etiquetax,etiquetay) {  
 ggscatter(Datos, x = colx, y = coly,  
 color = "black", shape = 21, size = 3,   
 title = title,  
 xlab = etiquetax,  
 ylab = etiquetay,  
 add = "reg.line",   
 add.params = list(color = "blue", fill = "lightgray"),   
 conf.int = TRUE,   
 cor.coef = TRUE,   
 cor.coeff.args = list(method = "pearson", label.x.npc = "center", label.y.npc = "top", label.sep = "\n")  
 )  
}  
  
  
  
###  
# Uso la funcion describeBy del package "psych" y me da todas los parametros descriptivos   
# agrupados por grupo  
###  
  
EstaDesc\_FertilityxGrupo <- t(describeBy(Datos$fertility,Datos$group,mat=TRUE))  
EstaDesc\_FertilityxGrupo

## X11 X12 X13   
## item "1" "2" "3"   
## group1 "africa" "oecd" "other"   
## vars "1" "1" "1"   
## n " 52" " 31" "110"   
## mean "4.272635" "1.765161" "2.360921"   
## sd "1.2888611" "0.3397387" "0.9409056"   
## median "4.4325" "1.7940" "2.1715"   
## trimmed "4.297167" "1.736760" "2.225043"   
## mad "1.3298922" "0.3825108" "0.7442652"   
## min "1.590" "1.312" "1.134"   
## max "6.925" "2.909" "5.968"   
## range "5.335" "1.597" "4.834"   
## skew "-0.2145928" " 1.0621651" " 1.5031817"  
## kurtosis "-0.7970165" " 1.7945323" " 2.6434380"  
## se "0.17873288" "0.06101888" "0.08971183"

EstaDesc\_FertilityxGrupo <- t(describeBy(Datos$ppgdp,Datos$group,mat=TRUE))  
EstaDesc\_FertilityxGrupo

## X11 X12 X13   
## item "1" "2" "3"   
## group1 "africa" "oecd" "other"   
## vars "1" "1" "1"   
## n " 52" " 31" "110"   
## mean " 2336.792" "37761.139" " 9818.937"  
## sd " 3423.781" "22092.070" "12502.384"  
## median " 891.25" "39545.90" " 4791.75"   
## trimmed " 1546.305" "35237.900" " 7080.803"  
## mad " 691.707" "19336.217" " 4979.979"  
## min " 114.8" "9100.7" " 499.0"   
## max " 16852.4" "105095.4" " 72397.9"   
## range "16737.6" "95994.7" "71898.9"   
## skew "2.427924" "0.964480" "2.292768"   
## kurtosis "5.937343" "1.051116" "5.966535"   
## se " 474.793" "3967.853" "1192.056"

#fertility group by group  
est\_descript <- descriptiva(Datos,(4:4),3)

## Note: Using an external vector in selections is ambiguous.  
## i Use `all\_of(colagruparpor)` instead of `colagruparpor` to silence this message.  
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This message is displayed once per session.

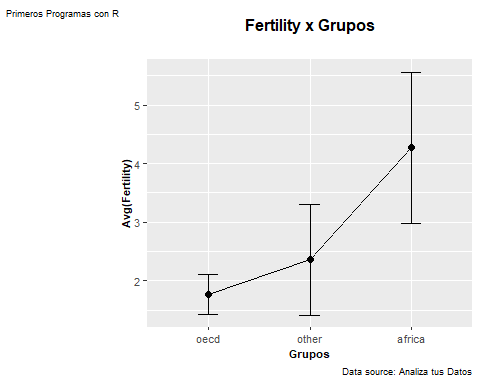
## Note: Using an external vector in selections is ambiguous.  
## i Use `all\_of(columnas)` instead of `columnas` to silence this message.  
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.  
## This message is displayed once per session.

est\_descript

## other africa oecd   
## Grupo "other" "africa" "oecd"   
## media "2.36092065947838" "4.27263461538462" "1.76516129032258"   
## cutmedia "2.27872806284094" "4.28027083333333" "1.74134482758621"   
## StdErr "0.0897118309306565" "0.1787328808768" "0.0610188776848853"  
## IC95LimInf "2.1850854708543" "3.92231816886609" "1.64556429006021"   
## IC95LimSup "2.53675584810247" "4.62295106190314" "1.88475829058496"   
## mediana "2.1715" "4.4325" "1.794"   
## sd "0.940905620656172" "1.2888611332254" "0.339738732633789"   
## var "0.885303386982377" "1.66116302073906" "0.115422406451613"   
## obs "110" "52" "31"   
## Min "1.134" "1.59" "1.312"   
## Max "5.968" "6.925" "2.909"   
## Rango "4.834" "5.335" "1.597"   
## RangoIQR "1.641" "3.18975" "1.4765"   
## Q1 "2.61625" "5.09225" "1.9485"   
## Q3 "0.97525" "1.9025" "0.472"   
## skew "1.50318167778327" "-0.214592799902107" "1.06216507953994"   
## kurtosis "2.6434380231697" "-0.797016491463271" "1.79453225589008"

crea\_errdiagram(Datos,"fertility","group","Fertility x Grupos","Grupos","Avg(Fertility)") #diagrama de error

## `summarise()` ungrouping output (override with `.groups` argument)

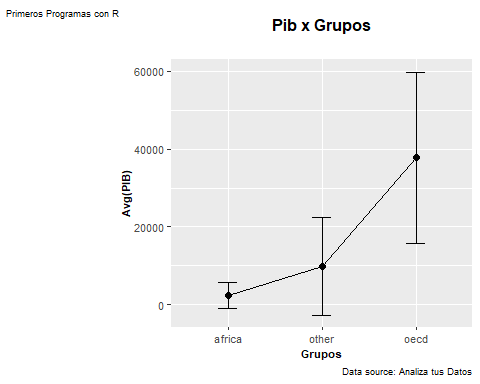


#ppgdp group by group  
est\_descript <- descriptiva(Datos,(5:5),3)   
est\_descript

## other africa oecd   
## Grupo "other" "africa" "oecd"   
## media "9818.93727272727" "2336.79230769231" "37761.1387096774"   
## cutmedia "8260.36" "1914.59583333333" "36427.5586206897"   
## StdErr "1192.05550090973" "474.792987214153" "3967.85285969731"   
## IC95LimInf "7482.50849094421" "1406.19805275257" "29984.1471046707"   
## IC95LimSup "12155.3660545103" "3267.38656263205" "45538.1303146841"   
## mediana "4791.75" "891.25" "39545.9"   
## sd "12502.3835686402" "3423.78092126269" "22092.0697491759"   
## var "156309594.897405" "11722275.7968024" "488059545.802452"   
## obs "110" "52" "31"   
## Min "499" "114.8" "9100.7"   
## Max "72397.9" "16852.4" "105095.4"   
## Rango "71898.9" "16737.6" "95994.7"   
## RangoIQR "2278.85" "494.45" "20138.2"   
## Q1 "10937.075" "2715.075" "46453.4"   
## Q3 "8658.225" "2220.625" "26315.2"   
## skew "2.29276752005822" "2.42792435299868" "0.964479978910561"  
## kurtosis "5.96653515494434" "5.93734309917983" "1.05111634614457"

crea\_errdiagram(Datos,"ppgdp","group","Pib x Grupos","Grupos","Avg(PIB)") #diagrama de error

## `summarise()` ungrouping output (override with `.groups` argument)

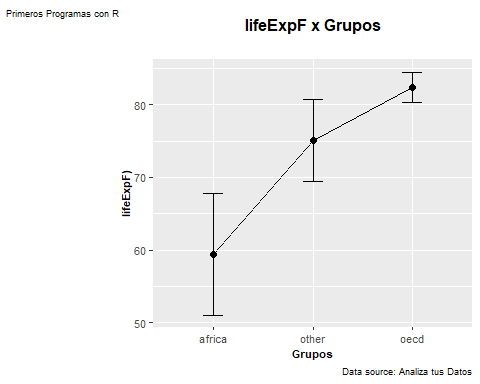


#lifeExpF group by group  
est\_descript <- descriptiva(Datos,(6:6),3)   
est\_descript

## other africa oecd   
## Grupo "other" "africa" "oecd"   
## media "75.1415951566926" "59.4217307692308" "82.4464516129032"   
## cutmedia "75.4518546723618" "59.1191666666667" "82.4865517241379"   
## StdErr "0.539698605995796" "1.1631321720076" "0.376271335518279"   
## IC95LimInf "74.0837858889408" "57.1419917120959" "81.7089597952874"   
## IC95LimSup "76.1994044244443" "61.7014698263657" "83.1839434305191"   
## mediana "76.1527336180905" "58.15" "82.79"   
## sd "5.66040673313488" "8.38746537263039" "2.09499013265313"   
## var "32.0402043845187" "70.3495753770739" "4.38898365591398"   
## obs "110" "52" "31"   
## Min "49.49" "48.11" "76.61"   
## Max "86.35" "77.86" "87.12"   
## Rango "36.86" "29.75" "10.51"   
## RangoIQR "72.4125" "53.0825" "81.34"   
## Q1 "78.1975" "63.34" "83.52"   
## Q3 "5.785" "10.2575" "2.17999999999999"   
## skew "-1.25599209512853" "0.849601944760743" "-0.631109235500854"  
## kurtosis "3.27751014862858" "-0.25409981249275" "0.671787451199418"

crea\_errdiagram(Datos,"lifeExpF","group","lifeExpF x Grupos","Grupos","lifeExpF)") #diagrama de error

## `summarise()` ungrouping output (override with `.groups` argument)

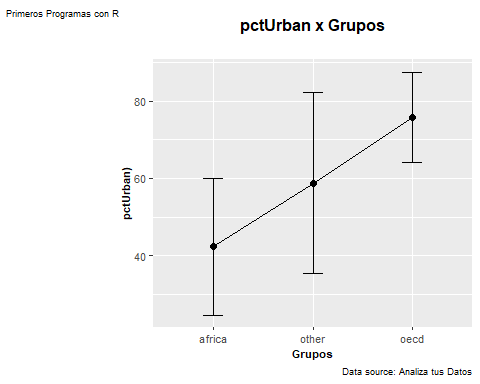


#pctUrban group by group  
est\_descript <- descriptiva(Datos,(7:7),3)   
est\_descript

## other africa oecd   
## Grupo "other" "africa" "oecd"   
## media "58.8" "42.3653846153846" "75.8064516129032"   
## cutmedia "58.9" "41.9791666666667" "76"   
## StdErr "2.23553827063332" "2.45595113192601" "2.10244064769317"   
## IC95LimInf "54.4183449895587" "37.5517203968096" "71.6856679434246"   
## IC95LimSup "63.1816550104413" "47.1790488339596" "79.9272352823819"   
## mediana "58.5" "39.5" "78"   
## sd "23.4465231866323" "17.7101154723861" "11.7058941131913"   
## var "549.739449541284" "313.648190045249" "137.027956989247"   
## obs "110" "52" "31"   
## Min "13" "11" "49"   
## Max "100" "86" "97"   
## Rango "87" "75" "48"   
## RangoIQR "41.75" "28" "68"   
## Q1 "75" "59" "85"   
## Q3 "33.25" "31" "17"   
## skew "-0.063562877871908" "0.347195579384427" "-0.344511009773028"  
## kurtosis "-0.913377266360203" "-0.586208023091504" "-0.788899883097915"

crea\_errdiagram(Datos,"pctUrban","group","pctUrban x Grupos","Grupos","pctUrban)") #diagrama de error

## `summarise()` ungrouping output (override with `.groups` argument)

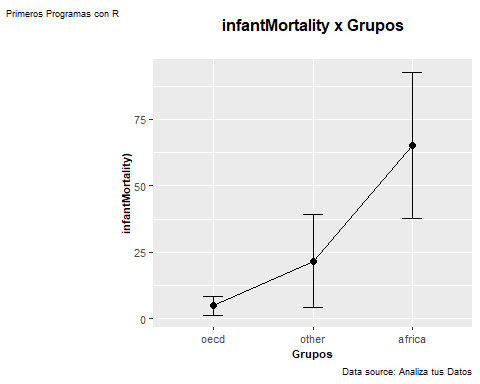


#infantMortality group by group  
est\_descript <- descriptiva(Datos,(8:8),3)   
est\_descript

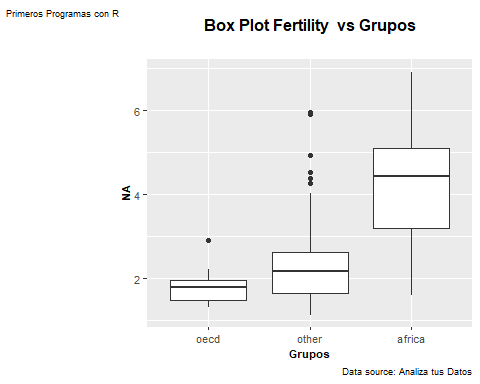
## other africa oecd   
## Grupo "other" "africa" "oecd"   
## media "21.676026405736" "65.3207692307692" "4.89009677419355"   
## cutmedia "20.1329090463096" "65.365875" "4.4621724137931"   
## StdErr "1.6664117632273" "3.81509536456634" "0.628788855196248"  
## IC95LimInf "18.4098593498105" "57.8431823162192" "3.6576706180089"   
## IC95LimSup "24.9421934616615" "72.7983561453193" "6.12252293037819"   
## mediana "18.268" "67.0175" "3.914"   
## sd "17.4774740196761" "27.5110439154579" "3.50094817970636"   
## var "305.462098108454" "756.85753731825" "12.2566381569892"   
## obs "110" "52" "31"   
## Min "1.916" "12.112" "2.289"   
## Max "124.535" "123.94" "19.901"   
## Rango "122.619" "111.828" "17.612"   
## RangoIQR "10.558" "46.936" "3.346"   
## Q1 "26.098" "85.88" "4.8415"   
## Q3 "15.54" "38.944" "1.4955"   
## skew "2.31164537550623" "-0.167533748188997" "3.07749155525433"   
## kurtosis "9.33196864985711" "-0.780644366833662" "9.52346826093656"

crea\_errdiagram(Datos,"infantMortality","group","infantMortality x Grupos","Grupos","infantMortality)") #diagrama de error

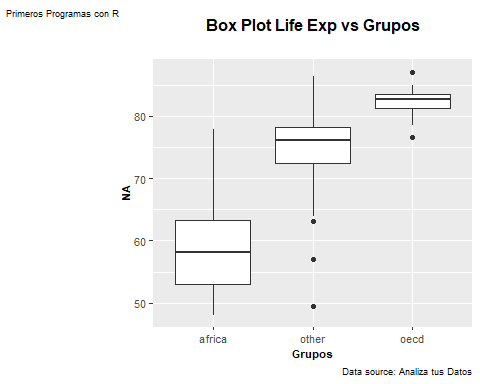
## `summarise()` ungrouping output (override with `.groups` argument)



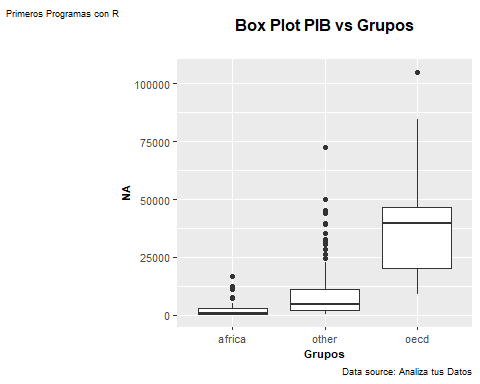
# Primeros Analisis  
  
##Box Plot para comparar grupos (comparacion de medias entre grupos)  
creaboxplot(Datos,"group","fertility","Box Plot Fertility vs Grupos","Grupos")



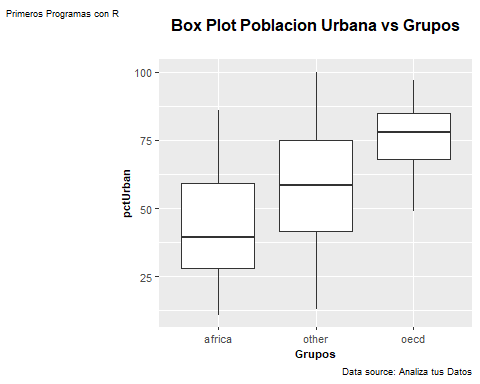
creaboxplot(Datos,"group","lifeExpF","Box Plot Life Exp vs Grupos","Grupos")



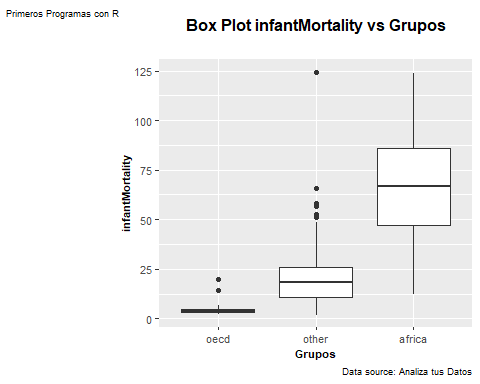
creaboxplot(Datos,"group","ppgdp","Box Plot PIB vs Grupos","Grupos")



creaboxplot(Datos,"group","pctUrban","Box Plot Poblacion Urbana vs Grupos","Grupos","pctUrban")

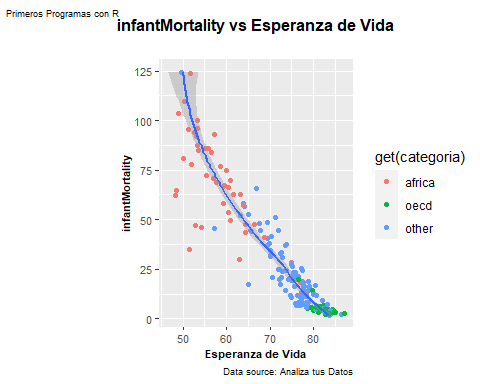


creaboxplot(Datos,"group","infantMortality","Box Plot infantMortality vs Grupos","Grupos","infantMortality")



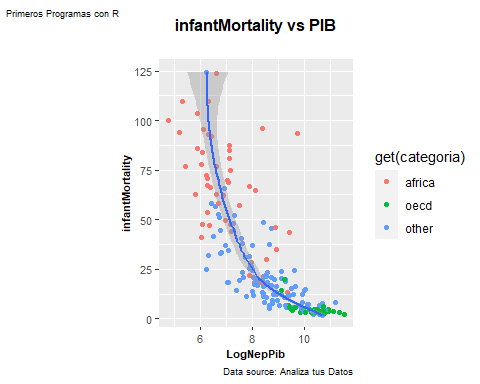
## Scatterplot para buscar asociaciones entre variables  
creascatplot(Datos,"lifeExpF","infantMortality","infantMortality vs Esperanza de Vida","Esperanza de Vida","infantMortality")

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



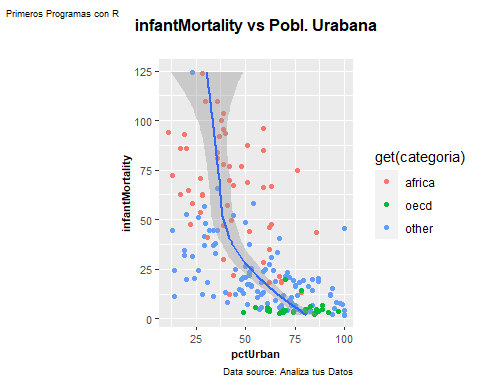
creascatplot(Datos,"LogNepPib","infantMortality","infantMortality vs PIB","LogNepPib","infantMortality")

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



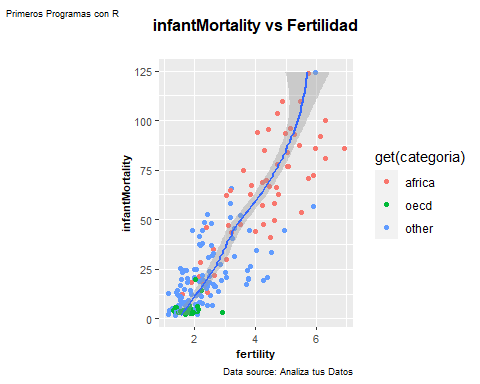
creascatplot(Datos,"pctUrban","infantMortality","infantMortality vs Pobl. Urabana","pctUrban","infantMortality")

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



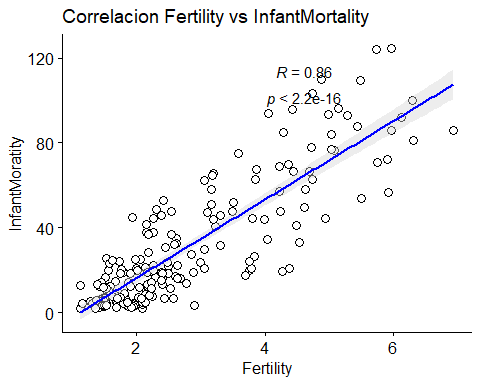
creascatplot(Datos,"fertility","infantMortality","infantMortality vs Fertilidad","fertility","infantMortality")

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



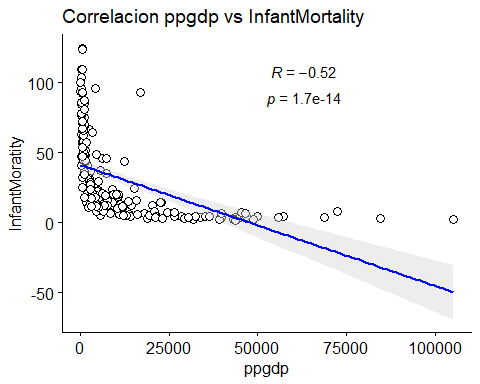
# Diagrama de Correlaciones  
creagrafcorr(Datos,"fertility","infantMortality","Correlacion Fertility vs InfantMortality","Fertility","InfantMoratity")

## `geom\_smooth()` using formula 'y ~ x'



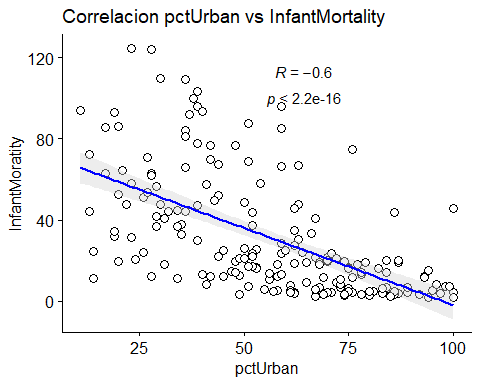
creagrafcorr(Datos,"ppgdp","infantMortality","Correlacion ppgdp vs InfantMortality","ppgdp","InfantMoratity")

## `geom\_smooth()` using formula 'y ~ x'



creagrafcorr(Datos,"pctUrban","infantMortality","Correlacion pctUrban vs InfantMortality","pctUrban","InfantMoratity")

## `geom\_smooth()` using formula 'y ~ x'



# Coef. de Correlacion InfanMortality vs variables  
print(paste("r de InfantMortality vs Fertilidad ", cor(Datos$infantMortality,Datos$fertility)))

## [1] "r de InfantMortality vs Fertilidad 0.857909640351146"

print(paste("r de InfantMortality vs Poblacion Urbana ", cor(Datos$infantMortality,Datos$pctUrban)))

## [1] "r de InfantMortality vs Poblacion Urbana -0.599161713229932"

print(paste("r de InfantMortality vs PIB ", cor(Datos$infantMortality,Datos$ppgdp)))

## [1] "r de InfantMortality vs PIB -0.515384695319901"

print(paste("r de InfantMortality vs Esperzana de Vida ", cor(Datos$infantMortality,Datos$lifeExpF)))

## [1] "r de InfantMortality vs Esperzana de Vida -0.933168259138956"

#Matriz de Correlacion  
  
res <- Datos %>% select(4:8)  
res <- cor(res)  
round(res, 2)

## fertility ppgdp lifeExpF pctUrban infantMortality  
## fertility 1.00 -0.44 -0.82 -0.54 0.86  
## ppgdp -0.44 1.00 0.56 0.59 -0.52  
## lifeExpF -0.82 0.56 1.00 0.60 -0.93  
## pctUrban -0.54 0.59 0.60 1.00 -0.60  
## infantMortality 0.86 -0.52 -0.93 -0.60 1.00