Taller 2 - Andrés Díaz, Yilmer Palacios

Andrés Díaz - Cod: 200610686, Yilmer Palacios - Cod: 202214473

2024-02-17

# Primer Punto

## Ustedes cuentan con varias bases de datos que cuenta con información del precio del Petróleo Brent, del Gas Natural, Carbón, Gasolina Motor Corriente y finalmente del IPC para Colombia desde el 2000 hasta el 2024 con una periodicidad mensual. Cada una de estas bases de datos toma el nombre del bien al que hacen referencia junto con el sufijo .csv. Por ejemplo, cuentan con “Carbón.csv” y así con los demás bienes. Cada una de estas bases cuenta con tres columnas, una primera columna con la fecha mensualizada, una segunda columna con el precio del bien para Colombia y una tercera columna con el precio del bien promedio en el mundo. En su trabajo como analistas de información se les solicita realizar un reporte sobre estas series. Para ello, su jefa les solicita realizar las siguientes tareas:

### 1.1. Establezcan un código que les permita cargar en el programa R las bases de datos de los diferentes precios de combustibles.

# Definimos directorio  
directorio <- getwd()  
setwd(directorio)  
  
# Cargamos datos  
  
IPC <- read.table("1.2.5.IPC\_Serie\_variaciones.csv", header = TRUE, sep = ",")  
  
#Separación de año y mes en nuevas columnas  
IPC$Año <- substr(IPC$Mes.Año, start = 1, stop = 4)   
IPC$Mes <- substr(IPC$Mes.Año, start = 5, stop = 7)  
IPC$Mes.Año2 <- paste("01", IPC$Mes, IPC$Año, sep="-")  
IPC$Mes.Año2 <- as.Date(IPC$Mes.Año2, "%d-%m-%Y")  
  
  
#Definición de formato  
IPC$Año <- as.numeric(IPC$Año)  
IPC$Mes <- as.numeric(IPC$Mes)  
  
IPC$Indice <- gsub(",",".",IPC$Indice)  
IPC$Indice <- as.numeric(IPC$Indice)  
  
  
precios\_carbon <- read.table("precios\_carbon.csv", header = TRUE,  
 sep = ",")  
precios\_carbon$fecha <- as.Date(precios\_carbon$fecha, "%Y-%m-%d")  
  
precios\_gas\_nat <- read.table("precios\_gas\_natural.csv",   
 header = TRUE, sep = ",")  
precios\_gas\_nat$fecha <- as.Date(precios\_gas\_nat$fecha, "%Y-%m-%d")  
  
precios\_gasolina <- read.table("precios\_gasolina.csv",   
 header = TRUE, sep = ",")  
precios\_gasolina$fecha <- as.Date(precios\_gasolina$fecha, "%Y-%m-%d")  
  
precios\_petroleo <- read.table("precios\_petroleo.csv",  
 header = TRUE, sep = ",")  
precios\_petroleo$fecha <- as.Date(precios\_petroleo$fecha, "%Y-%m-%d")

### 1.2. Realicen una exploración inicial de la base de datos. Para esto, para cada una de las bases de datos describan la misma. Es decir, redacten un breve párrafo comentando qué variables contiene la base de datos, el número de observaciones, el periodo de tiempo que comprende, el tipo de las variables, entre otros aspectos relevantes.

### Hacemos la descripción de la base de datos IPC  
cat("DESCRIPCION DE LA BASE DE DATOS IPC \n")

## DESCRIPCION DE LA BASE DE DATOS IPC

paste("La base de datos IPC tiene", length(IPC$Mes.Año2), "observaciones, su fecha inicia en ", min(IPC$Mes.Año2), "y termina en", max(IPC$Mes.Año2), "las variables y clases de variables son:")

## [1] "La base de datos IPC tiene 289 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:"

cat("\n")

str(IPC)

## 'data.frame': 289 obs. of 8 variables:  
## $ Mes.Año : int 202401 202312 202311 202310 202309 202308 202307 202306 202305 202304 ...  
## $ Indice : num 139 138 137 136 136 ...  
## $ Variación.anual.. : chr "8,35" "9,28" "10,15" "10,48" ...  
## $ Variación.mensual.. : chr "0,92" "0,45" "0,47" "0,25" ...  
## $ Variación.año.corrido..: chr "0,92" "9,28" "8,78" "8,27" ...  
## $ Año : num 2024 2023 2023 2023 2023 ...  
## $ Mes : num 1 12 11 10 9 8 7 6 5 4 ...  
## $ Mes.Año2 : Date, format: "2024-01-01" "2023-12-01" ...

cat("\n Otros datos relevantes son: \n")

##   
## Otros datos relevantes son:

summary(IPC)

## Mes.Año Indice Variación.anual.. Variación.mensual..  
## Min. :200001 Min. : 40.30 Length:289 Length:289   
## 1st Qu.:200601 1st Qu.: 59.02 Class :character Class :character   
## Median :201201 Median : 76.75 Mode :character Mode :character   
## Mean :201161 Mean : 79.17   
## 3rd Qu.:201801 3rd Qu.: 97.53   
## Max. :202401 Max. :138.98   
## Variación.año.corrido.. Año Mes Mes.Año2   
## Length:289 Min. :2000 Min. : 1.000 Min. :2000-01-01   
## Class :character 1st Qu.:2006 1st Qu.: 3.000 1st Qu.:2006-01-01   
## Mode :character Median :2012 Median : 6.000 Median :2012-01-01   
## Mean :2012 Mean : 6.481 Mean :2011-12-31   
## 3rd Qu.:2018 3rd Qu.: 9.000 3rd Qu.:2018-01-01   
## Max. :2024 Max. :12.000 Max. :2024-01-01

cat("\n \n")

### Hacemos la descripción de la base de datos Precios del Carbón  
cat("DESCRIPCION DE LA BASE DE DATOS PRECIOS DEL CARBÓN \n")

## DESCRIPCION DE LA BASE DE DATOS PRECIOS DEL CARBÓN

paste("La base de datos precios\_carbon tiene", length(precios\_carbon$fecha), "observaciones, su fecha inicia en ", min(precios\_carbon$fecha), "y termina en", max(precios\_carbon$fecha), "las variables y clases de variables son:")

## [1] "La base de datos precios\_carbon tiene 8667 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:"

cat("\n")

str(precios\_carbon)

## 'data.frame': 8667 obs. of 2 variables:  
## $ fecha : Date, format: "2000-01-01" "2000-01-02" ...  
## $ precio\_carbon: num 56.8 84.1 70.7 79.7 91.5 ...

cat("\n Otros datos relevantes son: \n")

##   
## Otros datos relevantes son:

summary(precios\_carbon)

## fecha precio\_carbon   
## Min. :2000-01-01 Min. : 50.98   
## 1st Qu.:2005-12-26 1st Qu.: 83.23   
## Median :2012-01-02 Median : 97.19   
## Mean :2011-12-31 Mean : 96.96   
## 3rd Qu.:2018-01-02 3rd Qu.:110.67   
## Max. :2024-01-01 Max. :142.97   
## NA's :88

cat("\n \n")

### Hacemos la descripción de la base de datos de Precios del Gas Natural  
cat("DESCRIPCION DE LA BASE DE DATOS GAS NATURAL \n")

## DESCRIPCION DE LA BASE DE DATOS GAS NATURAL

paste("La base de datos precios\_gas\_nat tiene", length(precios\_gas\_nat$fecha), "observaciones, su fecha inicia en ", min(precios\_gas\_nat$fecha), "y termina en", max(precios\_gas\_nat$fecha), "las variables y clases de variables son:")

## [1] "La base de datos precios\_gas\_nat tiene 8667 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:"

cat("\n")

str(precios\_gas\_nat)

## 'data.frame': 8667 obs. of 2 variables:  
## $ fecha : Date, format: "2000-01-01" "2000-01-02" ...  
## $ precio\_gas\_natural: num 134.1 102.8 72.7 101.5 133.8 ...

cat("\n Otros datos relevantes son: \n")

##   
## Otros datos relevantes son:

summary(precios\_gas\_nat)

## fecha precio\_gas\_natural  
## Min. :2000-01-01 Min. : 70.78   
## 1st Qu.:2005-12-30 1st Qu.:100.59   
## Median :2012-01-02 Median :118.36   
## Mean :2011-12-30 Mean :118.31   
## 3rd Qu.:2017-12-28 3rd Qu.:135.65   
## Max. :2024-01-01 Max. :165.38   
## NA's :88

cat("\n \n")

### Hacemos la descripción de la base de datos de Precios de la Gasolina  
cat("DESCRIPCION DE LA BASE DE DATOS PRECIOS DE LA GASOLINA \n")

## DESCRIPCION DE LA BASE DE DATOS PRECIOS DE LA GASOLINA

paste("La base de datos precios\_gasolina tiene", length(precios\_gasolina$fecha), "observaciones, su fecha inicia en ", min(precios\_gasolina$fecha), "y termina en", max(precios\_gasolina$fecha), "las variables y clases de variables son:")

## [1] "La base de datos precios\_gasolina tiene 8667 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:"

cat("\n")

str(precios\_gasolina)

## 'data.frame': 8667 obs. of 2 variables:  
## $ fecha : Date, format: "2000-01-01" "2000-01-02" ...  
## $ precio\_gasolina: num 85.2 58.8 62.4 48.6 43.8 ...

cat("\n Otros datos relevantes son: \n")

##   
## Otros datos relevantes son:

summary(precios\_gasolina)

## fecha precio\_gasolina   
## Min. :2000-01-01 Min. : 41.52   
## 1st Qu.:2006-01-03 1st Qu.: 87.12   
## Median :2012-01-02 Median :108.66   
## Mean :2012-01-01 Mean :108.94   
## 3rd Qu.:2017-12-28 3rd Qu.:130.73   
## Max. :2024-01-01 Max. :176.34   
## NA's :88

cat("\n \n")

### Hacemos la descripción de la base de datos de Precios del Petróleo  
cat("DESCRIPCION DE LA BASE DE DATOS PRECIOS DEL PETRÓLEO \n")

## DESCRIPCION DE LA BASE DE DATOS PRECIOS DEL PETRÓLEO

paste("La base de datos precios\_petroleo tiene", length(precios\_petroleo$fecha), "observaciones, su fecha inicia en ", min(precios\_petroleo$fecha), "y termina en", max(precios\_petroleo$fecha), "las variables y clases de variables son:")

## [1] "La base de datos precios\_petroleo tiene 8667 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:"

cat("\n")

str(precios\_petroleo)

## 'data.frame': 8667 obs. of 2 variables:  
## $ fecha : Date, format: "2000-01-01" "2000-01-02" ...  
## $ precio\_petroleo: num 77.7 108.4 118.8 105.7 115.3 ...

cat("\n Otros datos relevantes son: \n")

##   
## Otros datos relevantes son:

summary(precios\_petroleo)

## fecha precio\_petroleo   
## Min. :2000-01-01 Min. : 70.39   
## 1st Qu.:2005-12-30 1st Qu.: 95.55   
## Median :2011-12-24 Median :107.98   
## Mean :2011-12-28 Mean :108.09   
## 3rd Qu.:2017-12-27 3rd Qu.:120.71   
## Max. :2024-01-01 Max. :145.87   
## NA's :88

cat("\n \n")

#str(precios\_carbon)  
#summary(precios\_carbon)  
  
#str(precios\_gas\_nat)  
#summary(precios\_gas\_nat)  
  
#str(precios\_gasolina)  
#summary(precios\_gasolina)  
  
#str(precios\_petroleo)  
#summary(precios\_petroleo)

La base de datos **IPC** tiene 289 observaciones, su fecha inicia en 2000-01-01 y termina en 2024-01-01 las variables y clases de variables son:

1. Mes. Año: int
2. Indice: num
3. Variación.anual: chr
4. Variación.mensual: chr
5. Variación.año.corrido: chr
6. Año: num
7. Mes: num

Las bases de datos **precios\_carbon**, **precios\_gas\_nat**, **precios\_gasolina** y **precios\_petroleo** tienen cada una 8667 observaciones, sus fecha inician en 2000-01-01 y terminan en 2024-01-01, y sus variables y clases de variables son:

1. fecha: Date
2. precios\_carbon, precios\_gas\_nat, precios\_gasolina y precios\_petroleo: num

## Explorando las bases de datos de los precios de forma individual identifican que no todas tienen el mismo número de observaciones. Un hecho muy común cuando se trabaja con datos en formato panel es que algunas fechas para algunos productos no estén en la base de datos, es decir, hay fechas faltantes. A continuación, ustedes quieren “explicitar” las fechas faltantes.

### 1.3. Creen una función que tenga como parámetros un dataframe, una columna de fecha y una columna de precio, posteriormente, la función tiene que agregar las fechas faltantes en la columna de fecha y añadir un missing value en su valor correspondiente en el precio para esta fecha explicitada. Finalmente, tiene que retornar el dataframe con las fechas completas.

completar\_fechas <- function(base,fecha,precio){  
   
 #Input:  
 #base: dataframe de fechas y precios de combustibles  
 #fecha: columna de fechas  
 #precio: columna de valores numéricos con precio de combustibles  
   
 #Sacar fechas mínimas y máximas   
 fecha\_min <- min(fecha)  
 fecha\_max <- max(fecha)  
   
 #Crear vector de fechas completas  
 fecha\_completa <- fecha\_min:fecha\_max  
 fecha\_completa <- as.Date(fecha\_completa)  
  
 #Identificar cuáles fechas faltan  
 fechas\_faltantes <- fecha\_completa[which(!(fecha\_completa %in% fecha))]  
   
 #Crear vector de precios faltantes  
 precios\_faltantes <- rep(x = NA,length(fechas\_faltantes))  
   
 #Unir vectores de fechas y precios faltantes  
 base\_faltante <- as.data.frame(cbind(fechas\_faltantes,precios\_faltantes))  
   
 #Unir las bases incompletas  
 names(base\_faltante) <- names(base)  
 base <- rbind(base,base\_faltante)  
   
 #Organizar según fecha  
 base <- base[order(base$fecha),]  
   
 return(base)  
   
}

### 1.4. Apliquen la función anterior a cada uno de los precios en la base de datos.

precios\_carbon <- completar\_fechas(  
 precios\_carbon,precios\_carbon$fecha,precios\_carbon$precio\_carbon)  
  
precios\_gas\_nat <- completar\_fechas(  
 precios\_gas\_nat,precios\_gas\_nat$fecha,  
 precios\_gas\_nat$precio\_gas\_natural)  
  
precios\_gasolina <-completar\_fechas(  
 precios\_gasolina,precios\_gasolina$fecha,  
 precios\_gasolina$precio\_gasolina)  
  
precios\_petroleo <-completar\_fechas(  
 precios\_petroleo,precios\_petroleo$fecha,  
 precios\_petroleo$precio\_petroleo)

### 1.5. Realicen una unión de cada una de los datasets cargados que les permita juntar los precios y fechas en una misma base de datos.

precios\_combustible <- merge(precios\_carbon,precios\_gas\_nat,by = "fecha")  
precios\_combustible <- merge(precios\_combustible,precios\_gasolina,by = "fecha")  
precios\_combustible <- merge(precios\_combustible,precios\_petroleo,by = "fecha")  
  
head(precios\_combustible)

## fecha precio\_carbon precio\_gas\_natural precio\_gasolina precio\_petroleo  
## 1 2000-01-01 56.80856 134.10288 85.22463 77.7050  
## 2 2000-01-02 84.06858 102.80182 58.80204 108.3627  
## 3 2000-01-03 70.66739 72.73675 62.35808 118.7896  
## 4 2000-01-04 NA 101.51064 48.59105 105.6975  
## 5 2000-01-05 79.72284 133.78021 43.80961 115.3072  
## 6 2000-01-06 91.50991 78.58548 41.87021 100.5230

### 1.6. Realice las siguientes transformaciones de los datos: para la fecha transfórmenla de formato mm/dd/aa a un formato dd/mm/aa, revisen que las variables numéricas sean identificadas de tal forma y asegúrense de que tengan un formato de dos decimales.

precios\_combustible$fecha <- as.Date(precios\_combustible$fecha, "%d/%m/%Y")  
precios\_combustible$fecha <- format(precios\_combustible$fecha, "%d/%m/%Y")  
precios\_combustible[2:5] <- round(precios\_combustible[-1],2)  
head(precios\_combustible)

## fecha precio\_carbon precio\_gas\_natural precio\_gasolina precio\_petroleo  
## 1 01/01/2000 56.81 134.10 85.22 77.70  
## 2 02/01/2000 84.07 102.80 58.80 108.36  
## 3 03/01/2000 70.67 72.74 62.36 118.79  
## 4 04/01/2000 NA 101.51 48.59 105.70  
## 5 05/01/2000 79.72 133.78 43.81 115.31  
## 6 06/01/2000 91.51 78.59 41.87 100.52

### 1.7. Utilizando los operadores y funciones de la librería dplyr, creen respectivamente columnas que correspondan al mes y año de las observaciones. Filtren la base de datos para preservar únicamente las observaciones posteriores a enero del 2005 utilizando las columnas mes y año creadas anteriormente. Finalmente, eliminen de la base de datos las columnas asociadas a los precios de los combustibles promedio en el mundo.

#Incluir mes y año  
  
precios\_combustible$fecha <- as.Date(precios\_combustible$fecha, "%d/%m/%Y")  
  
precios\_combustible <- precios\_combustible %>%  
 mutate(Mes = as.numeric(format(fecha,"%m")))   
  
precios\_combustible <- precios\_combustible %>%  
 mutate(Año = as.numeric(format(fecha,"%Y")))  
  
#Filtrar la bases de datos  
  
precios\_combustible <- precios\_combustible %>% subset(Año>=2005) %>%   
 filter(!(Mes == 1 &   
 Año == 2005))  
  
head(precios\_combustible)

## fecha precio\_carbon precio\_gas\_natural precio\_gasolina precio\_petroleo  
## 1 2005-02-01 106.50 110.87 94.09 106.90  
## 2 2005-02-02 72.15 123.07 88.51 90.08  
## 3 2005-02-03 59.73 125.54 88.03 121.65  
## 4 2005-02-04 101.08 104.46 93.15 125.48  
## 5 2005-02-05 67.77 139.16 61.72 103.61  
## 6 2005-02-06 96.46 NA 105.22 80.32  
## Mes Año  
## 1 2 2005  
## 2 2 2005  
## 3 2 2005  
## 4 2 2005  
## 5 2 2005  
## 6 2 2005

### 1.8. Revisen ¿Qué porcentaje de la base de datos cuenta con valores faltantes en el precio de algún bien? Sustituyan estos valores faltantes con el precio promedio de ese bien.

#Lista con variables de precio  
combustibles\_p <- c("precio\_carbon","precio\_gas\_natural",  
 "precio\_gasolina","precio\_petroleo")  
  
#Estimación de cantidad de NA  
  
cantidad\_NA <- precios\_combustible[combustibles\_p] %>% is.na() %>% sum()  
cantidad\_sinNA <- sum(!is.na(precios\_combustible[combustibles\_p]))  
  
porcentaje <- cantidad\_NA/(cantidad\_NA+cantidad\_sinNA)  
  
print(paste("El porcentaje de NA es",round(porcentaje\*100,2),"%"))

## [1] "El porcentaje de NA es 2.13 %"

#Sustitución de los NA por el promedio  
  
precios\_combustible <- precios\_combustible %>%   
 mutate(precio\_carbon = ifelse(is.na(precio\_carbon),  
 mean(precio\_carbon,na.rm = TRUE),  
 precio\_carbon))  
  
precios\_combustible <- precios\_combustible %>%   
 mutate(precio\_gas\_natural = ifelse(is.na(precio\_gas\_natural),  
 mean(precio\_gas\_natural,na.rm = TRUE),  
 precio\_gas\_natural))  
  
precios\_combustible <- precios\_combustible %>%   
 mutate(precio\_gasolina = ifelse(is.na(precio\_gasolina),  
 mean(precio\_gasolina,na.rm = TRUE),  
 precio\_gasolina))  
  
precios\_combustible <- precios\_combustible %>%   
 mutate(precio\_petroleo = ifelse(is.na(precio\_petroleo),  
 mean(precio\_petroleo,na.rm = TRUE),  
 precio\_petroleo))  
  
#Verificación de imputación de valores  
  
print(paste("La cantidad de NA despúes de reemplaza los NA por el promedio es: ",sum(is.na(precios\_combustible))))

## [1] "La cantidad de NA despúes de reemplaza los NA por el promedio es: 0"

### El porcentaje de NA es 2.13 % antes del reemplazo

### 1.9. Agrupen la información de los precios a los valores promedios por mes y año. Por otra parte, para poder realizar comparaciones correctas entre precios es importante convertir los precios nominales en precios reales (constantes de un año base), pues permite controlar por el efecto inflacionario. Para transformarlo, se normaliza por la diferencia entre los Índices de Precios al Consumidor de la forma:

#### (1) 𝑃𝑟𝑒𝑎𝑙,𝑡 = 𝑃𝑛𝑜𝑚𝑖𝑛𝑎𝑙,𝑡 ∗ 𝐼𝑃𝐶𝑏𝑎𝑠𝑒/ 𝐼𝑃𝐶𝑡

### Donde 𝐼𝑃𝐶𝑏𝑎𝑠𝑒 hace referencia al IPC de un año-mes específico. 𝐼𝑃𝐶𝑡 al IPC del año-mes t sobre el cual se quiere convertir el precio nominal del año-mes t (𝑃𝑛𝑜𝑚𝑖𝑛𝑎𝑙,𝑡).

#Agrupar la información de los precios a valores promedios por mes y año  
  
tabla\_resumen <- precios\_combustible %>% group\_by(Año,Mes) %>%  
 summarise(precio\_carbon =   
 mean(precio\_carbon),  
 precio\_gas\_natural =   
 mean(precio\_gas\_natural),  
 precio\_gasolina =   
 mean(precio\_gasolina),  
 precio\_petroleo =   
 mean(precio\_petroleo))

## `summarise()` has grouped output by 'Año'. You can override using the `.groups`  
## argument.

head(tabla\_resumen)

## # A tibble: 6 × 6  
## # Groups: Año [1]  
## Año Mes precio\_carbon precio\_gas\_natural precio\_gasolina precio\_petroleo  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2005 2 82.5 113. 82.4 100.  
## 2 2005 3 83.1 111. 86.4 100.  
## 3 2005 4 90.2 104. 85.3 105.  
## 4 2005 5 86.5 112. 83.6 105.  
## 5 2005 6 83.2 106. 86.5 102.  
## 6 2005 7 85.9 108. 87.5 101.

### 1.10. Creen una función que tenga como parámetros: una columna de una serie de un precio de la base de datos, un año, un mes y la columna del IPC de Colombia. Posteriormente, la función debe crear una nueva columna cuyo nombre sea: el “nombre del bien” + “año base” + “mes base” + el sufijo “\_transformada”. Esta columna debe ser la transformación de valores nominales de la serie a valores reales con base en el año que toma como parámetro la función. Es decir, utilizando la ecuación (1) tendrán que consolidar una función que transforme los valores nominales en precios constantes con base en cualquier mes-año.

transf\_precios\_constantes <- function(precio,año,mes,IPC2){  
   
 #Input   
 #precio: lista (num) de precios del combustible  
 #año: año base (num) para cálculo de precios constantes [2000,2024]  
 #mes: mes base (num) para cálculo de precios constantes [1,12]  
 #IPC: lista de valores del índice (num)  
   
 tabla\_resumen2 <- tabla\_resumen  
 tabla\_resumen2 <- tabla\_resumen2 %>% mutate(Mes.Año =   
 paste(Año,Mes,sep =""))  
   
 IPC2 <- IPC2 %>% mutate(Mes.Año =   
 paste(Año,Mes,sep =""))  
   
 resumen\_conIPC <- merge(tabla\_resumen2,IPC2[,c("Mes.Año","Indice")],  
 by = "Mes.Año")  
   
 resumen\_conIPC <- resumen\_conIPC[order(resumen\_conIPC$Año,  
 resumen\_conIPC$Mes),]  
   
 #Estimación del precio real  
   
 IPC\_base <- resumen\_conIPC$Indice[which(resumen\_conIPC$Año == año & resumen\_conIPC$Mes == mes)]  
   
 IPC\_t <- resumen\_conIPC$Indice  
   
 resumen\_conIPC$precio\_real <- precio\*IPC\_base/IPC\_t  
   
 #Obtención del nombre de la columna  
 combust\_name <- deparse(substitute(precio)) #String de la columna  
   
 #Extracción del nombre del combustible  
 combust\_name <- substr(combust\_name,  
 nchar("tabla\_resumen2$precio\_"),  
 nchar("tabla\_resumen2$precio\_")+15)  
   
 #Nombre completo con combustible, año y mes  
 name\_columna <- paste(combust\_name,año,mes,"\_transformada")  
 names(resumen\_conIPC)[9] <- name\_columna  
 return(resumen\_conIPC[9])  
}

### 1.11. Apliquen la función que desarrolló en el literal anterior para todos los bienes que definió en la base de datos, pueden usar cualquier año-mes base, lo importante es que sean explícitos.

### Usaremos como base el IPC de noviembre de 2011  
  
añoDE <- 2012 # año base para el cálculo de precios reales  
mesDE <- 11 # mes base para el cálculo de precios reales  
  
dataexportada <- data.frame(tabla\_resumen[,"Año"])  
dataexportada [,2] <- data.frame(tabla\_resumen[,"Mes"])  
dataexportada [,3] <- data.frame(tabla\_resumen[,3])  
dataexportada [,4] <- transf\_precios\_constantes(tabla\_resumen$precio\_carbon,  
 añoDE,mesDE,IPC)  
  
  
dataexportada [,5] <- data.frame(tabla\_resumen[,4])  
dataexportada [,6] <- transf\_precios\_constantes(tabla\_resumen$precio\_gas\_natural,  
 añoDE,mesDE,IPC)  
  
  
dataexportada[,7] <- data.frame(tabla\_resumen[,5])  
dataexportada[,8] <- transf\_precios\_constantes(tabla\_resumen$precio\_gasolina,  
 añoDE,mesDE,IPC)  
  
  
dataexportada[,9] <- data.frame(tabla\_resumen[,6])  
dataexportada[,10] <- transf\_precios\_constantes(tabla\_resumen$precio\_petroleo,  
 añoDE,mesDE,IPC)  
  
head(dataexportada)

## Año Mes precio\_carbon carbon 2012 11 \_transformada precio\_gas\_natural  
## 1 2005 2 82.47500 112.7920 113.1861  
## 2 2005 3 83.05548 112.7161 110.7019  
## 3 2005 4 90.22567 121.8953 104.2077  
## 4 2005 5 86.50312 116.4023 111.7242  
## 5 2005 6 83.15356 111.4526 105.8310  
## 6 2005 7 85.88592 115.0556 108.4603  
## gas\_natural 2012 11 \_transformada precio\_gasolina  
## 1 154.7922 82.43143  
## 2 150.2356 86.38387  
## 3 140.7851 85.28292  
## 4 150.3409 83.63484  
## 5 141.8477 86.45458  
## 6 145.2970 87.48238  
## gasolina 2012 11 \_transformada precio\_petroleo petroleo 2012 11 \_transformada  
## 1 112.7324 100.0600 136.8411  
## 2 117.2331 100.4465 136.3177  
## 3 115.2176 104.5830 141.2921  
## 4 112.5426 104.5984 140.7520  
## 5 115.8771 101.9573 136.6558  
## 6 117.1942 100.9187 135.1939

### 1.12. Exporte la base de datos consolidada en un archivo .csv

write.csv(dataexportada, "./dataconsolidada.csv")

# Punto 2

## A partir de la base de datos consolidada en el punto anterior, este punto busca explorar y analizar los datos consolidados. Para esto, realicen lo siguiente:

### 2.1. Presenten una tabla de estadísticas descriptivas donde evidencien el número de observaciones, promedio, min, max, desviación estándar de los precios nominales y reales de los combustibles. Expliquen e interpreten en el documento los valores más importantes de la tabla.

dataconsolidada <- read.table("dataconsolidada.csv", header = TRUE, sep = ",")  
describe(dataconsolidada[,4:11])

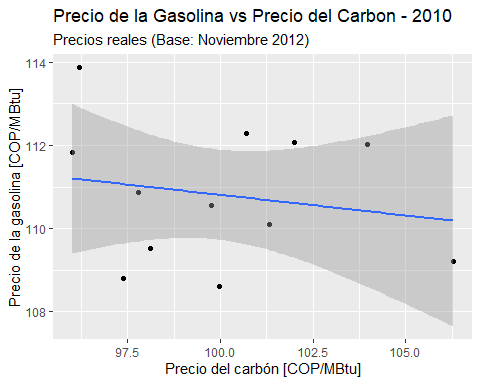
## vars n mean sd median trimmed mad  
## precio\_carbon 1 228 101.50 9.99 102.06 101.54 11.98  
## carbon.2012.11.\_transformada 2 228 93.60 12.69 95.68 93.89 14.62  
## precio\_gas\_natural 3 228 121.03 6.74 121.19 121.10 7.50  
## gas\_natural.2012.11.\_transformada 4 228 112.88 20.70 114.96 113.06 23.38  
## precio\_gasolina 5 228 118.43 20.08 118.10 118.31 24.98  
## gasolina.2012.11.\_transformada 6 228 107.38 7.24 109.04 108.22 6.26  
## precio\_petroleo 7 228 110.83 6.34 111.30 110.91 7.18  
## petroleo.2012.11.\_transformada 8 228 103.19 18.15 105.38 103.17 19.43  
## min max range skew kurtosis se  
## precio\_carbon 80.83 123.16 42.33 -0.03 -1.04 0.66  
## carbon.2012.11.\_transformada 57.17 121.90 64.73 -0.21 -0.52 0.84  
## precio\_gas\_natural 98.71 137.15 38.44 -0.15 -0.31 0.45  
## gas\_natural.2012.11.\_transformada 55.38 154.79 99.41 -0.09 -0.74 1.37  
## precio\_gasolina 82.43 166.10 83.67 0.07 -1.12 1.33  
## gasolina.2012.11.\_transformada 85.22 120.70 35.48 -1.05 1.02 0.48  
## precio\_petroleo 97.79 128.58 30.79 -0.06 -0.75 0.42  
## petroleo.2012.11.\_transformada 59.37 141.29 81.92 0.01 -0.74 1.20

#### Como se puede apreciar en la tabla, todas las variables de precios reales tienen medias cercanas al 110, siendo la media del carbón la más barata y como es de esperarse la de la gasolina la más cara; por otro lado, el precio real del gas natural fue el que tuvo mayor desviación estándar en su precio, aunque no es una medida de volatilidad o varianza directa si nos muestra que su rango de precios fue más extenso, es posible que en algunos periodos la volatilidad o varianza haya sido más severa que en otras series; finalmente la tabla nos muestra que caso contrario al precio real del gas natural, los precios reales de la gasolina tuvieron menor desviación estándar presentando mayor estabilidad con el paso del tiempo.

### 2.2. Presenten una gráfica de dispersión de puntos (scatter) donde en el eje X esté el precio del carbón y en el eje Y el precio de la gasolina, ambos reales sobre un año determinado, añadan una línea de ajuste lineal, título, labels y demás elementos que permitan una gráfica autocontenida. Expliquen e interpreten en el documento los elementos más importantes de la base de datos.

#Tomamos el año 2010 para estudio  
  
# Creamos una columna llamada fecha y le damos clase Date  
dataconsolidada$fecha <- as.character(paste0(dataconsolidada$Año,"-", dataconsolidada$Mes,"-","01"))  
dataconsolidada$fecha <- as.Date(dataconsolidada$fecha, "%Y-%m-%d")  
  
minaño <- "2010-01-01" #fecha menor del intervalo a graficar  
maxaño <- "2011-01-01" #fecha mayor del intervalo a graficar  
serieañodisper <- dataconsolidada[dataconsolidada[,12]>=minaño & dataconsolidada[,12]< maxaño,]  
  
serieXnombre <- serieañodisper[,5]  
serieYnombre <- serieañodisper[,9]  
  
  
ggplot(data = serieañodisper, aes(x = serieXnombre, y = serieYnombre)) +  
 geom\_point() + geom\_smooth(method = "lm") + labs(title = "Precio de la Gasolina vs Precio del Carbon - 2010", subtitle = "Precios reales (Base: Noviembre 2012)",  
 x = "Precio del carbón [COP/MBtu]",   
 y = "Precio de la gasolina [COP/MBtu]")

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

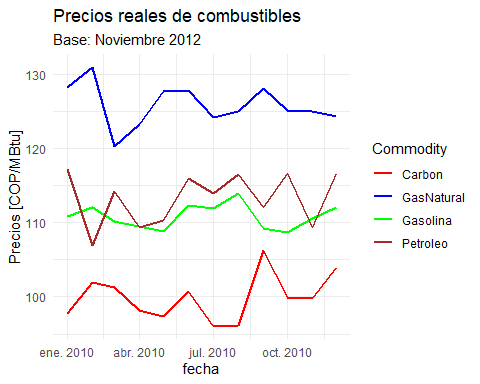


#### Para 2010, podemos apreciar que la relación lineal entre los precios reales del carbón y de la gasolina es negativa, sin realizar pruebas de hipótesis pareciera que, cuando el precio del carbón sube, baja el de la gasolina; sin embargo, la pendiente es casi horizontal lo cual indica que tienen una correlación débil, es decir que el comportamiento del carbón pareciera no estar correlacionado con el de la gasolina. En caso de querer realizar un análisis más robusto, se recomienda realizar una regresión lineal entre ambos commodities y realizar pruebas de hipótesis (asumiendo que se cumplen supuestos de series estacionarias).

### 2.3. Presenten una gráfica de serie de tiempo con el precio real del Carbón, Gas Natural, Petróleo y Gasolina Corriente para un mismo año, asegúrense de que tenga el título, labels, colores y demás elementos que permitan una gráfica autocontenida. Expliquen e interpreten en el documento los elementos más importantes de la base de datos.

# Creamos una columna llamada fecha y le damos clase Date  
dataconsolidada$fecha <- as.character(paste0(dataconsolidada$Año,"-", dataconsolidada$Mes,"-","01"))  
dataconsolidada$fecha <- as.Date(dataconsolidada$fecha, "%Y-%m-%d")  
  
# Creamos data frames para cada commodity luego los unimos en un solo dataframe  
PRcarbon <- data.frame(dataconsolidada$fecha)  
PRcarbon[,2] <- dataconsolidada[,5]  
PRcarbon[,3] <- "Carbon"  
  
PRgasnat <- data.frame(dataconsolidada$fecha)  
PRgasnat[,2] <- dataconsolidada[,7]  
PRgasnat[,3] <- "GasNatural"  
  
PRgasolina <- data.frame(dataconsolidada$fecha)  
PRgasolina[,2] <- dataconsolidada[,9]  
PRgasolina[,3] <- "Gasolina"  
  
PRPetroleo <- data.frame(dataconsolidada$fecha)  
PRPetroleo[,2] <- dataconsolidada[,11]  
PRPetroleo[,3] <- "Petroleo"  
  
serietiempo <- rbind(PRcarbon, PRgasnat, PRgasolina, PRPetroleo)  
colnames(serietiempo) <- c("fecha", "Precio", "Commodity")  
  
  
# Acotamos la serie al año 2010  
  
minaño <- "2010-01-01" #fecha menor del intervalo a graficar  
maxaño <- "2011-01-01" #fecha mayor del intervalo a graficar  
serietiempo2 <- serietiempo[serietiempo[,1:3]>=minaño & serietiempo[,1:3] <maxaño,1:3]  
  
  
# Graficamos  
ggplot(serietiempo2, aes(x = fecha, y = Precio)) +   
 geom\_line(aes(color = Commodity), size = 1) +  
 scale\_color\_manual(values = c("red", "blue", "green", "brown")) +  
 theme\_minimal()+ labs(title = "Precios reales de combustibles",  
 subtitle = "Base: Noviembre 2012",  
 x = "fecha", y = "Precios [COP/MBtu]")

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.



2.4. Realicen una visualización de tipo “heatmap” donde puedan identificar el cambio mensual del precio del Carbón, Petróleo y Gas Natural para cada uno de los años

#install.packages("ggpubr")  
library(ggpubr)

## Warning: package 'ggpubr' was built under R version 4.3.3

Meses <- seq(2,12,by = 2)  
  
#Precio carbón  
mapa\_carbon <- ggplot(dataconsolidada,aes(Año, Mes,   
 fill = carbon.2012.11.\_transformada))+  
 geom\_tile()+  
   
 labs(fill = "Precio real [COP/MBtu]")+  
 scale\_fill\_gradient(low="yellow", high="red")+  
 theme\_bw()+  
 scale\_y\_discrete(name ="Mes",   
 limits=Meses)

## Warning in scale\_y\_discrete(name = "Mes", limits = Meses): Continuous limits supplied to discrete scale.  
## ℹ Did you mean `limits = factor(...)` or `scale\_\*\_continuous()`?

#Precio gas natural  
mapa\_gas\_n <- ggplot(dataconsolidada,aes(Año, Mes,   
 fill = gas\_natural.2012.11.\_transformada))+  
 geom\_tile()+  
   
 labs(fill = "Precio real [COP/MBtu]")+  
 scale\_fill\_gradient(low="yellow", high="red")+  
 theme\_bw()+  
 scale\_y\_discrete(name ="Mes",   
 limits=Meses)

## Warning in scale\_y\_discrete(name = "Mes", limits = Meses): Continuous limits supplied to discrete scale.  
## ℹ Did you mean `limits = factor(...)` or `scale\_\*\_continuous()`?

#Precio gasolina  
mapa\_gasolina <- ggplot(dataconsolidada,aes(Año, Mes,   
 fill = gasolina.2012.11.\_transformada))+  
 geom\_tile()+  
   
 labs(fill = "Precio real [COP/MBtu]")+  
 scale\_fill\_gradient(low="yellow", high="red")+  
 theme\_bw()+  
 scale\_y\_discrete(name ="Mes",   
 limits=Meses)

## Warning in scale\_y\_discrete(name = "Mes", limits = Meses): Continuous limits supplied to discrete scale.  
## ℹ Did you mean `limits = factor(...)` or `scale\_\*\_continuous()`?

#Precio petroleo  
mapa\_petroleo <- ggplot(dataconsolidada,aes(Año, Mes,   
 fill = petroleo.2012.11.\_transformada))+  
 geom\_tile()+  
   
 labs(fill = "Precio real [COP/MBtu]")+  
 scale\_fill\_gradient(low="yellow", high="red")+  
 theme\_bw()+  
 scale\_y\_discrete(name ="Mes",   
 limits=Meses)

## Warning in scale\_y\_discrete(name = "Mes", limits = Meses): Continuous limits supplied to discrete scale.  
## ℹ Did you mean `limits = factor(...)` or `scale\_\*\_continuous()`?

mapa\_comb<- ggarrange(  
 "","",mapa\_carbon, mapa\_gas\_n, "","",mapa\_gasolina, mapa\_petroleo,  
 ncol = 2, nrow = 4,  
 labels = c("Panel A: Precio del carbón",   
 "Panel B: Precio del gas natural",  
 "","",  
 "Panel C: Precio de la gasolina",  
 "Panel D: Precio del petroleo"),  
 common.legend = TRUE, legend = "bottom", heights = c(1,7,1,7),  
 font.label=list(color="black",size=10)  
)  
  
  
 mapa\_comb

