

Análisis de preguntas relacionadas con recetarios oficiales

Misión PRI 1901

true

30-07-2021

Abstract

Análisis de preguntas relacionadas con recetarios oficiales. Se realiza una revisión de la segunda parte de la herramienta de recolección a Fondos Rotatorio de Medicamentos.

```
require(plotly)
require(tidyverse); theme_set(theme_bw())
require(lubridate)
require(ggrepel)
require(patchwork)

source(file.path('src', 'data', '901_funcionesMapa.R'), encoding = 'UTF-8')
source(file.path('src', 'models', '900_funcionesAlmacenamientoGrafico.R'), encoding = 'UTF-8')
```

```
# 1. Lectura de datos base -----
```

```
#
df <- read_csv(file.path('data', 'processed', '001_Herramienta_Procesada.csv'))

## Rows: 16 Columns: 184
## -- Column specification -----
## Delimiter: ","
## chr (158): Nombre del encuestador, Nombre del funcionario que recibe la enc...
## dbl (16): 3.01 Existencias actuales de recetarios en el FRE, 3.03 Tiempo d...
## lgl (7): Si la respuesta a la pregunta anterior fue "otro", indique cual:...
## dtm (3): Marca temporal, Fecha de la visita, Fecha creación FRE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
df_DIVIPOLA <-
  read_csv(file.path('data', 'processed', '798_DANE_DEPARTAMENTO.csv'))

## Rows: 33 Columns: 2
## -- Column specification -----
## Delimiter: ","
## chr (1): NombreDepartamento
## dbl (1): CodigoDepartamento
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
df_MUNICIPIO <-
  read_csv(file.path('data', 'processed', '799_DANE_DIVIPOLA.csv'),
            locale = locale(encoding = 'latin1'))

## Rows: 1121 Columns: 7

## -- Column specification -----
## Delimiter: ","
## chr (4): CODIGO_DEPARTAMENTO, NOMBRE_DEPARTAMENTO, NOMBRE_MUNICIPIO, TIPO
## dbl (3): CODIGO_MUNICIPIO, LATITUD, LONGITUD

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
df$`3.08 Precio de venta por prescripción (COP)`

## [1] 660.00 799.00 605.68 811.00 1200.00 400.00 500.00 1530.00 240.00
## [10] 606.00 300.00 1119.00 500.00 1160.00 568.00 300.00

df$CodigoDepartamento

## [1] "23" "08" "05" "47" "13" "54" "41" "85" "73" "63" "15" "76" "95" "86" "52"
## [16] "19"

df_total <- df %>%
  right_join(colombiaGeoDF, by = c('CodigoDepartamento' = 'DPTO'))

df_total

## # A tibble: 33 x 189
##   `Marca temporal`   `Nombre del encue~` `Nombre del funci~` `Nombre del funcio~
##   <dtm>             <chr>             <chr>             <chr>
## 1 2021-05-08 15:55:12 Carlos Guillermo ~ Heliodoro Kerguel~ Erick Novoa
## 2 2021-05-10 08:38:47 Wilder Estiben Me~ Jovana Vitola Jul~ Leidy Llorente Vil~
## 3 2021-05-10 12:03:39 Carlos Guillermo ~ Paola Andrea Góme~ <NA>
## 4 2021-05-21 15:23:40 Wilder Estiben Me~ Rafael Tapia Buen~ <NA>
## 5 2021-05-23 15:36:50 Carlos Guillermo ~ Maria Eugenia Bar~ <NA>
## 6 2021-05-25 21:56:53 Jhonathan Venegas Amilcar Marquez <NA>
## 7 2021-05-26 11:17:20 Nicolás Alexander~ Diana Carolina Va~ Edimer Covaleda Yu~
## 8 2021-05-26 11:59:27 Nicolás Alexander~ Julieta Palacios ~ Jonathan Sánchez M~
## 9 2021-05-26 13:01:15 Nicolás Alexander~ Marly Julieth Esp~ <NA>
## 10 2021-05-28 12:07:29 Wilder Estiben Me~ Adriana María Mur~ <NA>
## # ... with 23 more rows, and 185 more variables: Fecha de la visita <dtm>,
## # Departamento <chr>, Ciudad <chr>, Dirección <chr>, E-mail <chr>,
## # Telefono <chr>, Acto administrativo de creación <chr>,
## # Fecha creación FRE <dtm>, Nombre del Secretario de Salud <chr>,
## # Nombre del Gobernador <chr>, Nombre del funcionario a cargo del FRE <chr>,
## # Profesión del funcionario a cargo del FRE <chr>,
## # Si la respuesta a la pregunta anterior fue "otro", indique cual:...17 <chr>, ...
```

```
# 2. Número de existencias actuales en el FRE-----
```

```

labExistencias <- c('Sin existencias', '< 1000 u', '1000-5000 u', '> 5000 u')

df_total <- df_total %>%
  mutate(ExistenciasRecetarios = case_when(
    `3.01 Existencias actuales de recetarios en el FRE` == 0 ~ labExistencias[1],
    `3.01 Existencias actuales de recetarios en el FRE` < 1000 ~ labExistencias[2],
    (`3.01 Existencias actuales de recetarios en el FRE` >= 1000) &
      (`3.01 Existencias actuales de recetarios en el FRE` < 5000) ~ labExistencias[3],
    `3.01 Existencias actuales de recetarios en el FRE` >= 5000 ~ labExistencias[4]
  ))

gRecetarios1 <- df_total %>%
  mutate(ExistenciasRecetarios = factor(ExistenciasRecetarios,
                                         levels = labExistencias)) %>%

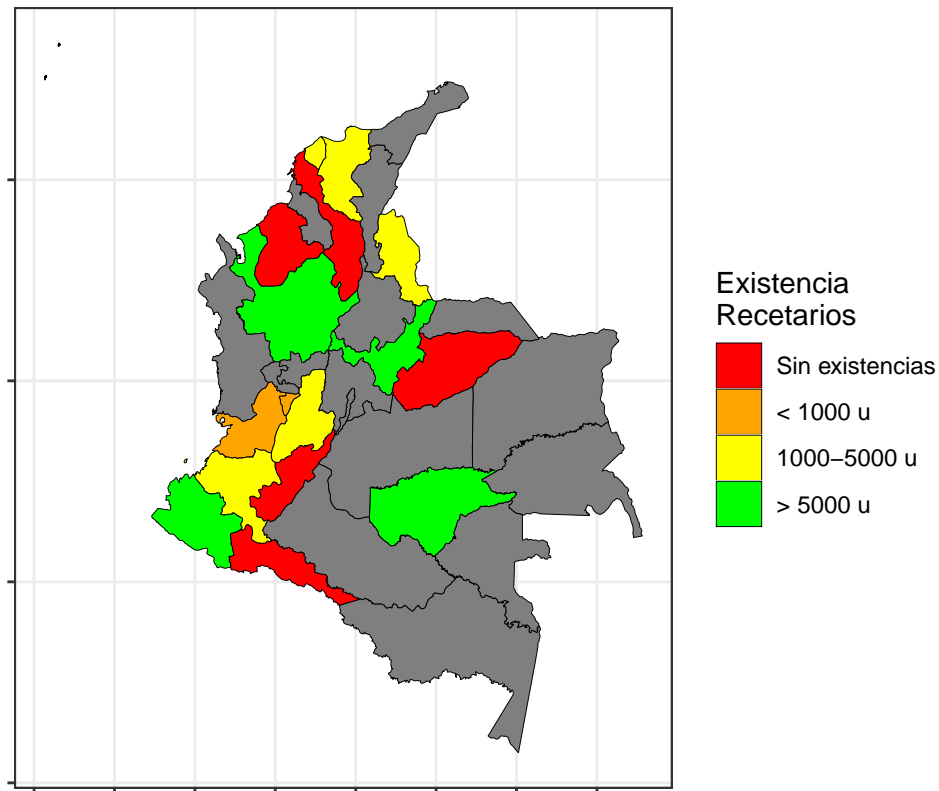
  ggplot() +
  geom_sf(
    aes(fill = ExistenciasRecetarios, geometry = geometry),
    colour = 'black',
    size = 0.1
  ) +
  # coord_sf(crs = st_crs(32618)) +
  theme(legend.position = 'right',
        axis.text = element_blank()) +
  labs(title = 'Existencias de recetarios en el FRE') +
  scale_fill_manual(name = 'Existencia \nRecetarios',
                    values = c('red', 'orange', 'yellow', 'green') %>%
                      setNames(., labExistencias)) +

  guides(
    fill = guide_legend(title.position = 'top'))

gRecetarios1

```

Existencias de recetarios en el FRE



```
guardarGGplot(gRecetarios1, '027_existenciaRecetarios_1', 8, 6)
```

3. Existencias en circulación en el departamento-----

```
promedioCirculacionRecetarios <- function(string) {
  variable <- tryCatch({
    string1 <- as.double(string)
  }, warning = function(w) {
    string1 <- str_split(string, '\\-') %>% map_dbl(~mean(as.double(.x)))
  }, error = function(e) {
    string1 <- str_split(string, '\\-') %>% map_dbl(~mean(as.double(.x)))
  })

  variable
}

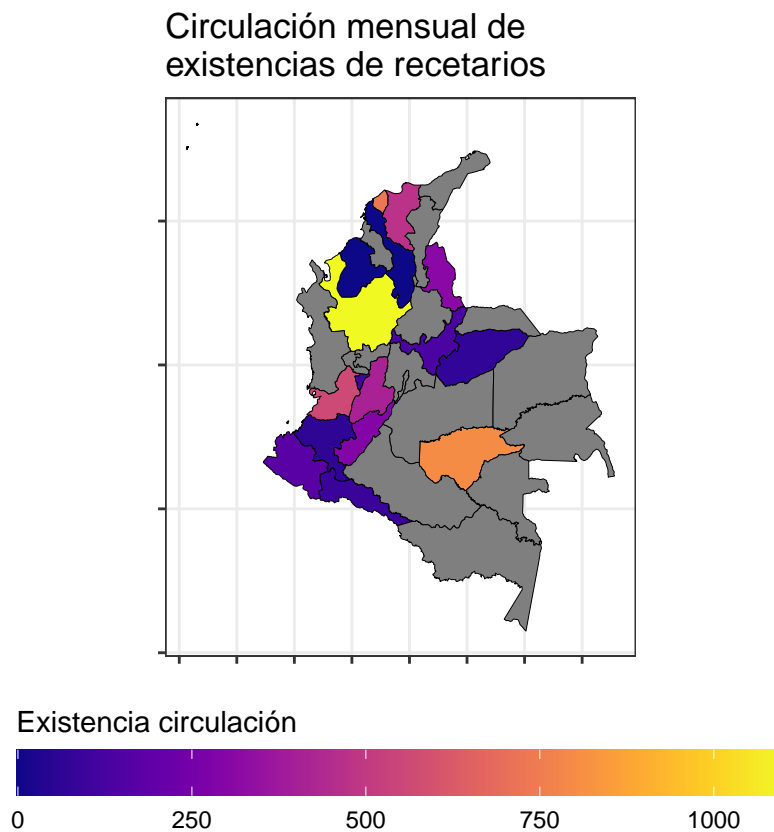
df_total <- df_total %>%
  mutate(
    ExistenciasCirculacion = map_dbl(
      `3.02 Existencias estimadas de recetarios en circulación en el departamento (promedio mensual)`,
      ~ promedioCirculacionRecetarios(.x)
    )
  )
```

```

gRecetarios2 <- df_total %>%
  ggplot() +
  geom_sf(
    aes(fill = ExistenciasCirculacion, geometry = geometry),
    colour = 'black',
    size = 0.1
  ) +
  paletteer::scale_fill_paletteer_c("viridis::plasma", name = 'Existencia circulación') +
  # coord_sf(crs = st_crs(32618)) +
  theme(legend.position = 'bottom',
        axis.text = element_blank()) +
  labs(title = 'Circulación mensual de \nexistencias de recetarios') +
  guides(
    fill = guide_colourbar(barwidth = 20, title.position = 'top'))

gRecetarios2

```



```

guardarGGplot(gRecetarios2, '028_existenciaRecetarios_2', 7, 5)

```

4. Duración estimada de las existencias -----

```

gRecetarios3 <- df_total %>%
  ggplot() +

```

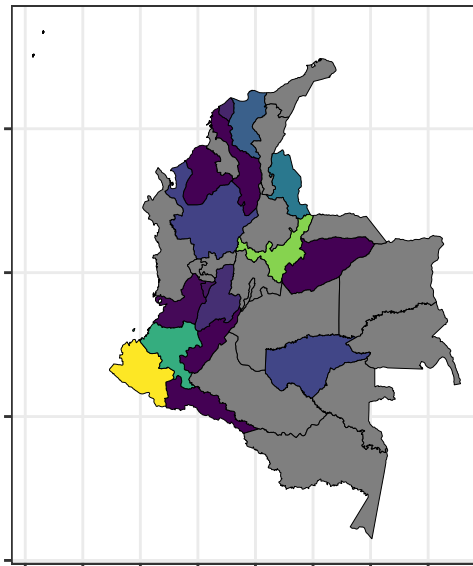
```

geom_sf(
  aes(fill = `3.03 Tiempo de duración proyectada de las actuales existencias de recetarios (semanas).`,
    geometry = geometry),
  colour = 'black',
  size = 0.1
) +
scale_fill_continuous(type = 'viridis', name = 'Duración recetarios') +
# coord_sf(crs = st_crs(32618)) +
theme(legend.position = 'bottom',
  axis.text = element_blank()) +
labs(title = 'Duración de existencias de recetarios \n(semanas)') +
guides(
  fill = guide_colourbar(barwidth = 20, title.position = 'top'))

```

gRecetarios3

Duración de existencias de recetarios (semanas)



Duración recetarios



```
guardarGGplot(gRecetarios3, '029_existenciaRecetarios_3', 7, 5)
```

5. Número de prescripciones por recetario -----

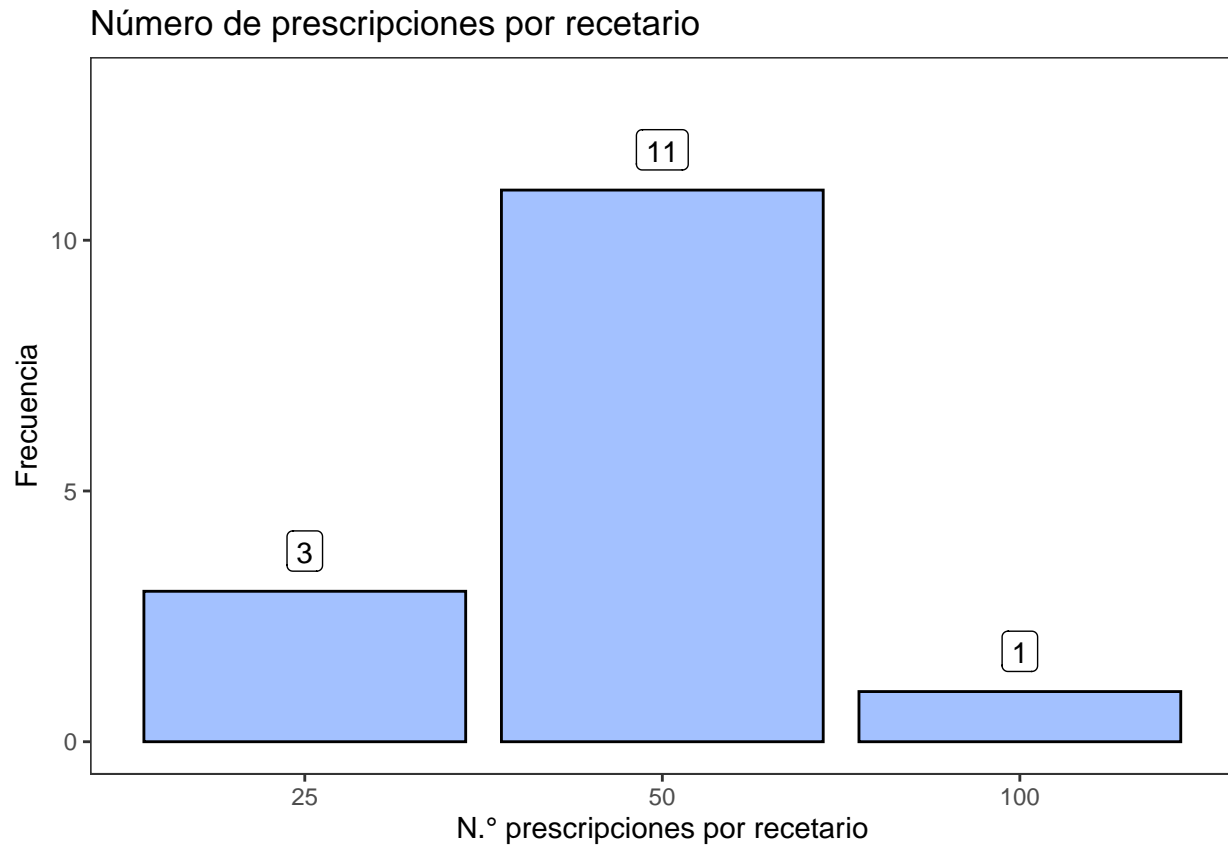
```

gRecetarios4 <- df_total %>%
  select(col1 = `3.05 N.º de prescripciones por recetario`) %>%
  ungroup() %>%

```

```
drop_na() %>%
ggplot(aes(x = factor(col1))) +
geom_bar(stat = 'count', fill = '#6699ff', color = 'black', alpha = 0.6) +
geom_label(aes(label = ..count..), stat = 'count', vjust = -0.5) +
coord_cartesian(ylim = c(0, 13)) +
ylab('Frecuencia') + xlab('N.º prescripciones por recetario') +
labs(title = 'Número de prescripciones por recetario') +
theme(panel.grid = element_blank())
```

gRecetarios4



```
guardarGGplot(gRecetarios4, '030_existenciaRecetarios_4', 6, 4)
```

6. Costo de adquisición por recetarios-----

```
gCostoRecetario <- df_total %>%
ggplot() +
geom_sf(
aes(fill = `3.06 Costo de adquisición del recetario (COP)`,
geometry = geometry),
colour = 'black',
size = 0.1
) +
```

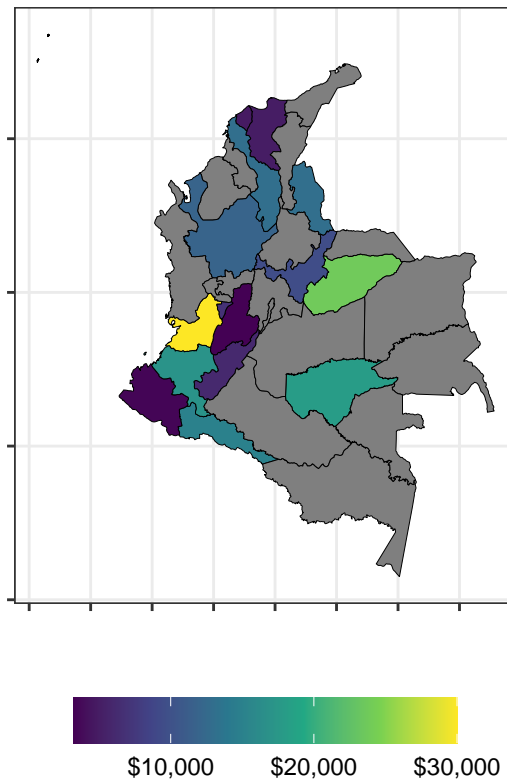
```

scale_fill_continuous(label = scales::dollar, type = 'viridis', ) +
# geom_sf_label_repel(aes(label = `3.06 Costo de adquisición del recetario (COP)`)) +
# coord_sf(crs = st_crs(32618)) +
theme(legend.position = 'bottom',
      axis.text = element_blank()) +
labs(title = 'Costo de adquisición') +
guides(
  fill = guide_colourbar(barwidth = 10, title.position = 'top', title = ''))

```

gCostoRecetario

Costo de adquisición



```

gPVTARecetario <- df_total %>%
  ggplot() +
  geom_sf(
    aes(fill = `3.07 Precio de venta del recetario (COP)`,
        geometry = geometry),
    colour = 'black',
    size = 0.1
  ) +
  scale_fill_continuous(label = scales::dollar, type = 'viridis') +
# geom_sf_label_repel(aes(label = `3.06 Costo de adquisición del recetario (COP)`)) +
# coord_sf(crs = st_crs(32618)) +
theme(legend.position = 'bottom',
      axis.text = element_blank()) +
labs(title = 'Precio de venta') +
guides(

```



```

    fill = guide_colourbar(barwidth = 10, title.position = 'top', title = ''))

(gCostoRecetario + gPVTARecetario) %>%
guardarGGplot(., '031_costoRecetario', 6, 4)

df_total <- df_total %>%
  mutate(MargenGanancia = map2_dbl(`3.06 Costo de adquisición del recetario (COP)`,
    `3.07 Precio de venta del recetario (COP)`, ~((.y - .x)/.x)))

meanMargenGanancia <- df_total$MargenGanancia %>% mean(na.rm= TRUE)

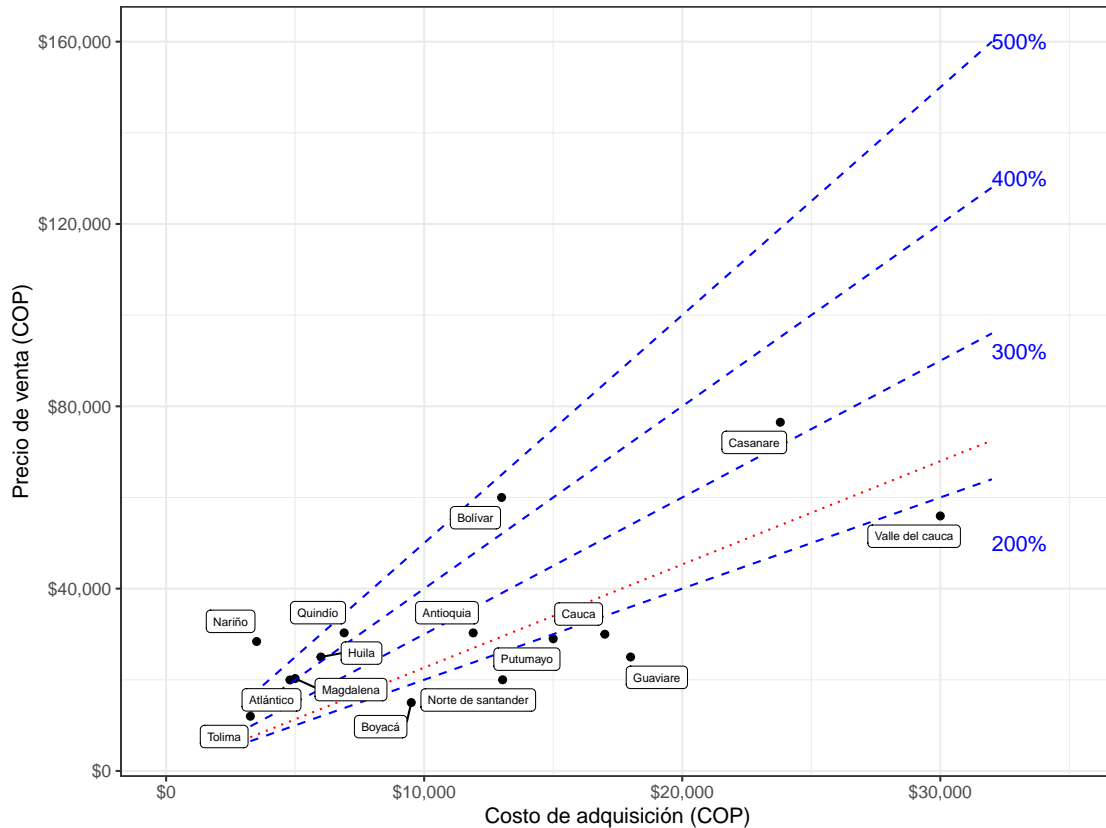
posicion1Lineas <- data.frame(
  label = c('200%', '300%', '400%', '500%'),
  xpos = rep(32000, 4),
  ypos = c(5e4, 9.2e4, 1.3e5, 1.6e5)
)

gComparativo1 <- df_total %>%
  mutate(Departamento_1 = str_to_sentence(Departamento_1)) %>%
  ggplot(aes(x = `3.06 Costo de adquisición del recetario (COP)`,
    y = `3.07 Precio de venta del recetario (COP)`) +
    geom_point() +
    stat_function(fun = function(x) 2*x, geom = 'line', linetype = 'dashed', col = 'blue') +
    stat_function(fun = function(x) 3*x, geom = 'line', linetype = 'dashed', col = 'blue') +
    stat_function(fun = function(x) 4*x, geom = 'line', linetype = 'dashed', col = 'blue') +
    stat_function(fun = function(x) 5*x, geom = 'line', linetype = 'dashed', col = 'blue') +
    stat_function(fun = function(x) meanMargenGanancia*x, geom = 'line', linetype = 'dotted', col = 'red') +
    geom_text(data = posicion1Lineas, aes(x = xpos, y = ypos, label = label), hjust = 0, col = 'blue') +
    geom_label_repel(aes(label = Departamento_1), size = 2, max.overlaps = Inf) +
    coord_cartesian(xlim = c(0, 35000)) +
    scale_x_continuous(labels = scales::dollar) +
    scale_y_continuous(labels = scales::dollar) +
    xlab('Costo de adquisición (COP)') +
    ylab('Precio de venta (COP)')
)

gComparativo1

## Warning: Removed 18 rows containing missing values (geom_point).
## Warning: Removed 18 rows containing missing values (geom_label_repel).

```



```
guardarGGplot(gComparativo1, '032_comparativoDepartamentos', 8, 6)
```

```
## Warning: Removed 18 rows containing missing values (geom_point).
```

```
## Warning: Removed 18 rows containing missing values (geom_label_repel).
```

```
# ¿El margen de ganancia de recetarios se corresponde con el número de personas en el FRE?
```

```
df_total <- df_total %>% rowwise() %>%
```

```
  mutate(Profesiones = list(c_across(matches('Profesión\\sdel\\s(personal|funcionario)'))))
```

```
df_total$NoPersonas <- df_total$Profesiones %>%
```

```
  map_dbl(function(x){sum(!is.na(x))})
```

```
gComparativo2 <- df_total %>%
```

```
  mutate(Departamento_1 = str_to_sentence(Departamento_1)) %>%
```

```
  ggplot(aes(x = NoPersonas, y = MargenGanancia)) +
```

```
  geom_point()+
```

```
  geom_label_repel(aes(label = Departamento_1)) +
```

```
  scale_y_continuous(labels = scales::percent) +
```

```
  ylab('Margen de ganancia por recetarios (%)') +
```

```
  xlab('N.º de personas')
```

```
gComparativo3 <- df_total %>%
```

```
  ggplot(aes(x = factor(NoPersonas), y = MargenGanancia)) +
```

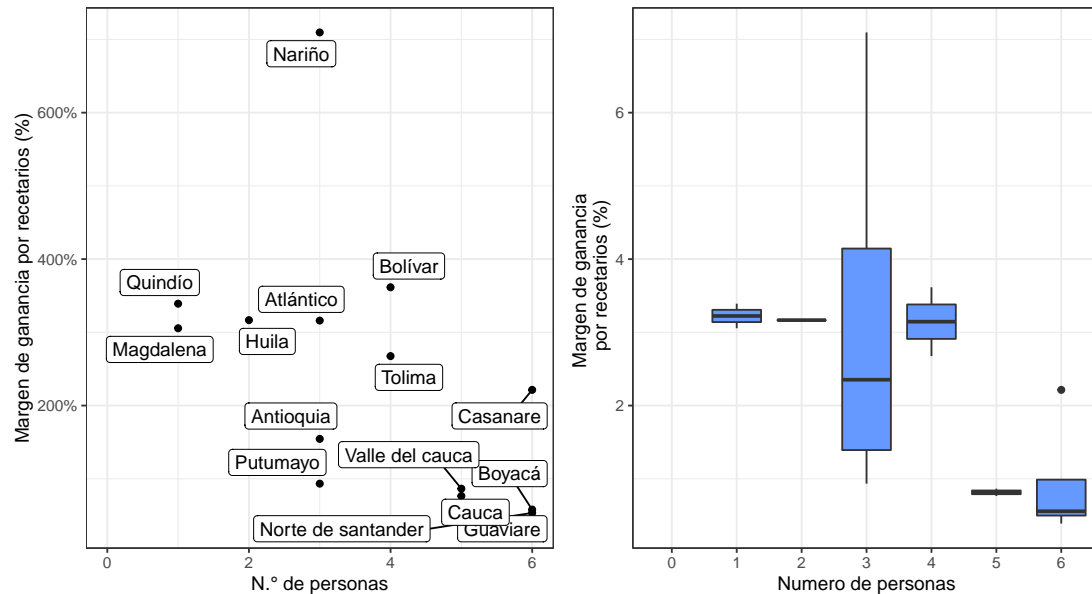
```
  geom_boxplot(fill = '#6699ff') +
```

```
  xlab('Numero de personas') +
```

```
  ylab('Margen de ganancia \n por recetarios (%)')
```

```
(gComparativo2 + gComparativo3)
```

```
## Warning: Removed 18 rows containing missing values (geom_point).
## Warning: Removed 18 rows containing missing values (geom_label_repel).
## Warning: Removed 18 rows containing non-finite values (stat_boxplot).
```



```
(gComparativo2 + gComparativo3) %>%
  guardarGGplot('033_comparativoDepartamentos1', 9, 5)
```

```
## Warning: Removed 18 rows containing missing values (geom_point).
## Warning: Removed 18 rows containing missing values (geom_label_repel).
## Warning: Removed 18 rows containing non-finite values (stat_boxplot).
```

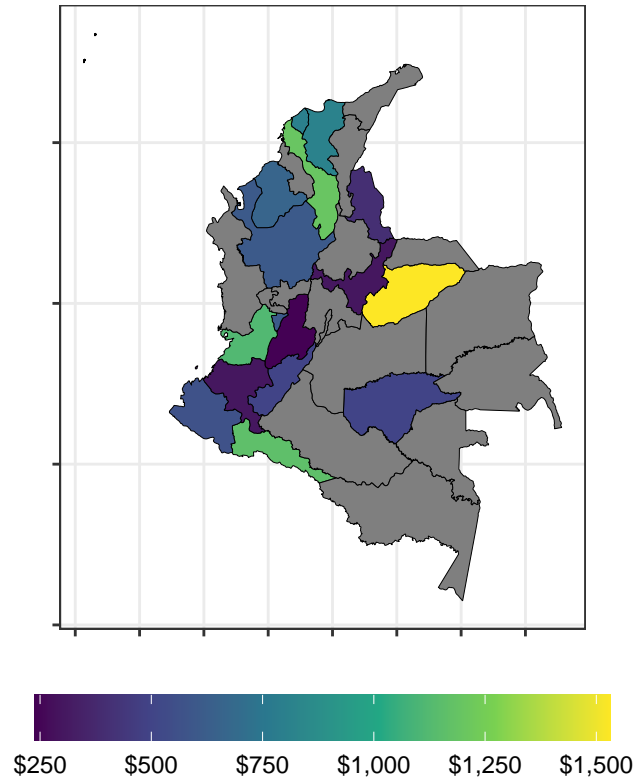
7. Precio de venta por prescripción-----

```
gPVTARecetario7 <- df_total %>%
  ggplot() +
  geom_sf(
    aes(fill = `3.08 Precio de venta por prescripción (COP)`,
        geometry = geometry),
    colour = 'black',
    size = 0.1
  ) +
  scale_fill_continuous(label = scales::dollar, type = 'viridis', name = NULL) +
  # geom_sf_label_repel(aes(label = `3.06 Costo de adquisición del recetaio (COP)`)) +
  # coord_sf(crs = st_crs(32618)) +
  theme(legend.position = 'bottom',
        axis.text = element_blank()) +
  labs(title = 'Precio de venta de recetas por prescripción') +
  guides(
```

```
fill = guide_colourbar(barwidth = 15, title.position = 'top'))
```

```
gPVTARecetario7
```

Precio de venta de recetarios por prescripción



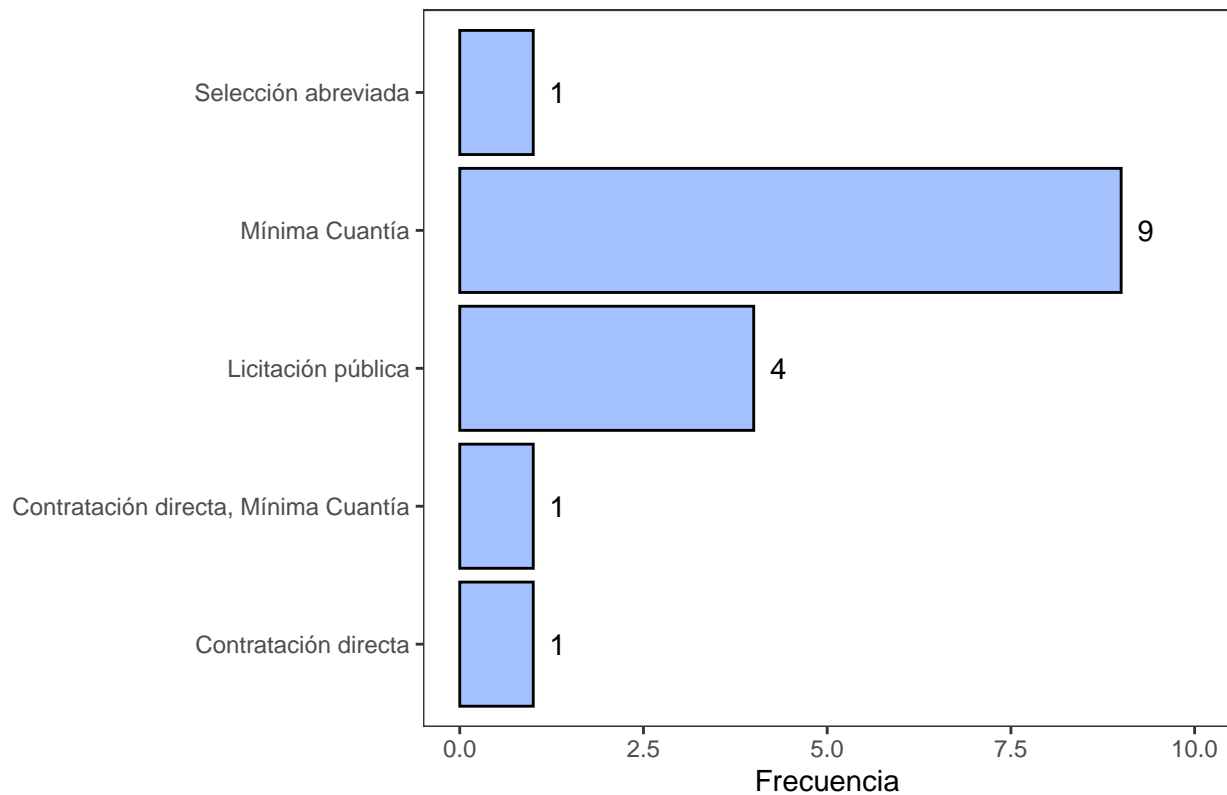
```
guardarGGplot(gPVTARecetario7, '034_PVTA_recetarios_prescripcion', 7, 4)
```

8. Modalidades de contratación para adquisición de Recetarios-----

```
gModalidadAdquisicion <- df_total %>%
  pull(`3.13. ¿Qué modalidades de selección se utilizan en la contratación para adquisición de recetarios?`) %>%
  table() %>% as_tibble() %>%
  ggplot(aes(y = ., x = n)) +
  geom_bar(stat = 'identity', fill = '#6699ff', color = 'black', alpha = 0.6) +
  geom_text(aes(label = n), position = position_dodge(width = 1),
    hjust = -1) +
  xlab('Frecuencia') +
  coord_cartesian(xlim = c(0, 10)) +
  labs(title = 'Modalidad de Selección Contratación Recetarios') +
  theme(axis.title.y = element_blank(),
    panel.grid = element_blank())
```

```
gModalidadAdquisicion
```

Modalidad de Selección Contratación Recetarios



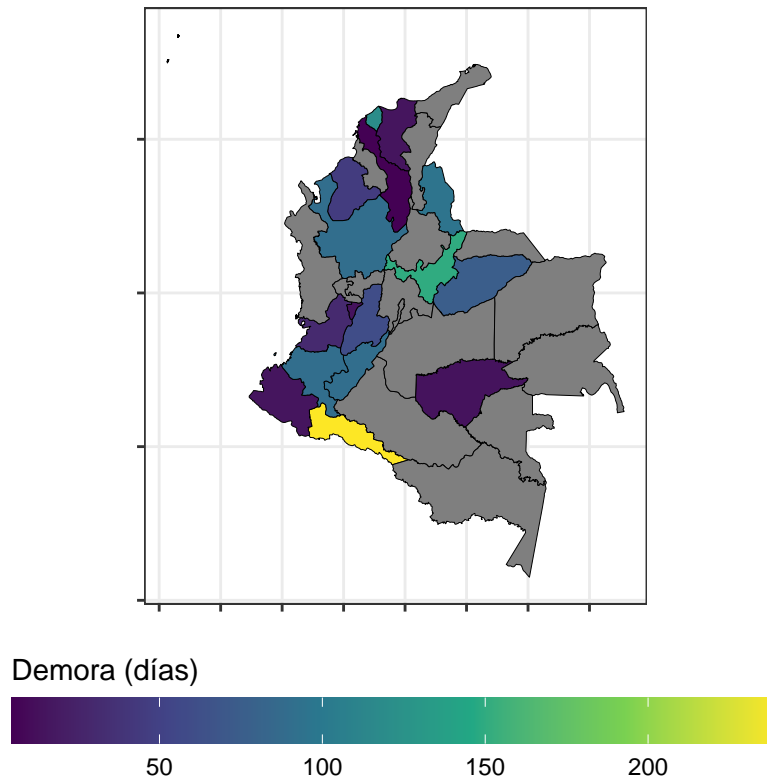
```
guardarGGplot(gModalidadAdquisicion, '035_modalidadAdquisicion', 7, 4)
```

9. Tiempo de demora para adquisición de recetarios -----

```
gDemoraRecetario <- df_total %>%
  ggplot() +
  geom_sf(
    aes(fill = `3.16. ¿Cuánto tiempo toma la adquisición de recetarios? (días)`,
        geometry = geometry),
    colour = 'black',
    size = 0.1
  ) +
  scale_fill_continuous(type = 'viridis') +
  # geom_sf_label_repel(aes(label = `3.06 Costo de adquisición del recetario (COP)`)) +
  # coord_sf(crs = st_crs(32618)) +
  theme(legend.position = 'bottom',
        axis.text = element_blank()) +
  labs(title = 'Tiempo de demora para adquisición de recetarios') +
  guides(
    fill = guide_colourbar(barwidth = 20, title.position = 'top', title = 'Demora (días)')
  )

gDemoraRecetario
```

Tiempo de demora para adquisición de recetarios



```
guardarGGplot(gDemoraRecetario, '036_demoraRecetario', 7, 5)
```