

Práctica de Evaluación FAD - Métodos de Análisis de Datos

Isabela Ignacio, Luisa Yáñez, Miguel García

18/12/2021

0. Introducción

La práctica consiste en la elaboración y presentación de un informe de un proyecto de Ciencia de Datos, utilizando las técnicas aprendidas durante el curso, aplicadas a los datos seleccionados.

1. Uso de herramienta de control de versiones

El grupo eligió trabajar en lenguaje R (RStudio version 1.4.1717) y utilizar como herramienta de control de versiones Github. El proyecto “/practica_fd_final” fue creado por Luisa Yáñez (usuario lyanezgu) y compartido con los restantes participantes del grupo Isabela Ignacio (usuario IsaPires1329) y Miguel García (usuario mgarciasanc2021).

2. Conjunto de datos elegido

El conjunto de datos elegido por el grupo se llama “Hospital Charges in America” y incluye información que compara las tarifas de los servicios de hospitalización en diferentes estados de los EEUU para los 100 principales grupos de diagnósticos.

Link del data set: <https://www.kaggle.com/dhirajnirne/hospital-charges-in-america>.

2.1 Paquetes

```
library(formatR)
library(readr)
library(ggplot2)
library(GGally)
library(dplyr)
library(tidyr)
library(missForest)
library(VIM)
library(formattable)
library(usmap)
library(cowplot)
```

2.2 Cargar los datos

El conjunto de datos “Hospital Charges in America” contiene 12 columnas y 163.065 filas, y lo obtenemos en formato .csv. Inicialmente se han guardado los datos en un data frame llamado “hospital_charges” y se ha realizado un estudio inicial sobre su contenido utilizando la función head y summary.

```
hospital_charges <- read_csv("notebooks/hospital-charges.csv")  
hospital_charges
```

```
## # A tibble: 163,065 x 12  
##   'DRG Definition' 'Provider Id' 'Provider Name' 'Provider Street ~  
##   <chr>           <dbl> <chr>          <chr>  
## 1 039 - EXTRACRANIAL PRO~ 10001 SOUTHEAST ALABAMA M~ 1108 ROSS CLARK C~  
## 2 039 - EXTRACRANIAL PRO~ 10005 MARSHALL MEDICAL CE~ 2505 U S HIGHWAY ~  
## 3 039 - EXTRACRANIAL PRO~ 10006 ELIZA COFFEE MEMORI~ 205 MARENGO STREET~  
## 4 039 - EXTRACRANIAL PRO~ 10011 ST VINCENT'S EAST    50 MEDICAL PARK E~  
## 5 039 - EXTRACRANIAL PRO~ 10016 SHELBY BAPTIST MEDI~ 1000 FIRST STREET~  
## 6 039 - EXTRACRANIAL PRO~ 10023 BAPTIST MEDICAL CEN~ 2105 EAST SOUTH B~  
## 7 039 - EXTRACRANIAL PRO~ 10029 EAST ALABAMA MEDICA~ 2000 PEPPERELL PA~  
## 8 039 - EXTRACRANIAL PRO~ 10033 UNIVERSITY OF ALABA~ 619 SOUTH 19TH ST~  
## 9 039 - EXTRACRANIAL PRO~ 10039 HUNTSVILLE HOSPITAL 101 SIVLEY RD  
## 10 039 - EXTRACRANIAL PRO~ 10040 GADSDEN REGIONAL ME~ 1007 GOODYEAR AVE~  
## # ... with 163,055 more rows, and 8 more variables: Provider City <chr>,  
## #   Provider State <chr>, Provider Zip Code <dbl>,  
## #   Hospital Referral Region Description <chr>, Total Discharges <dbl>,  
## #   Average Covered Charges <chr>, Average Total Payments <chr>,  
## #   Average Medicare Payments <chr>
```

```
head(hospital_charges)
```

```
## # A tibble: 6 x 12  
##   'DRG Definition' 'Provider Id' 'Provider Name' 'Provider Street ~  
##   <chr>           <dbl> <chr>          <chr>  
## 1 039 - EXTRACRANIAL PROC~ 10001 SOUTHEAST ALABAMA M~ 1108 ROSS CLARK C~  
## 2 039 - EXTRACRANIAL PROC~ 10005 MARSHALL MEDICAL CE~ 2505 U S HIGHWAY ~  
## 3 039 - EXTRACRANIAL PROC~ 10006 ELIZA COFFEE MEMORI~ 205 MARENGO STREET~  
## 4 039 - EXTRACRANIAL PROC~ 10011 ST VINCENT'S EAST    50 MEDICAL PARK E~  
## 5 039 - EXTRACRANIAL PROC~ 10016 SHELBY BAPTIST MEDI~ 1000 FIRST STREET~  
## 6 039 - EXTRACRANIAL PROC~ 10023 BAPTIST MEDICAL CEN~ 2105 EAST SOUTH B~  
## # ... with 8 more variables: Provider City <chr>, Provider State <chr>,  
## #   Provider Zip Code <dbl>, Hospital Referral Region Description <chr>,  
## #   Total Discharges <dbl>, Average Covered Charges <chr>,  
## #   Average Total Payments <chr>, Average Medicare Payments <chr>
```

```
summary(hospital_charges)
```

```
## DRG Definition      Provider Id      Provider Name      Provider Street Address  
## Length:163065      Min.   : 10001      Length:163065      Length:163065  
## Class :character   1st Qu.:110092     Class :character   Class :character  
## Mode  :character   Median :250007     Mode  :character   Mode  :character  
##                           Mean   :255570  
##                           3rd Qu.:380075
```

```

##          Max.    :670077
## Provider City      Provider State      Provider Zip Code
## Length:163065      Length:163065      Min.    : 1040
## Class  :character  Class  :character  1st Qu.:27261
## Mode   :character  Mode   :character  Median  :44309
##                               Mean    :47938
##                               3rd Qu.:72901
##                               Max.    :99835
## Hospital Referral Region Description Total Discharges  Average Covered Charges
## Length:163065          Min.    : 11.00  Length:163065
## Class  :character      1st Qu.: 17.00  Class  :character
## Mode   :character      Median : 27.00  Mode   :character
##                               Mean    : 42.78
##                               3rd Qu.: 49.00
##                               Max.    :3383.00
## Average Total Payments Average Medicare Payments
## Length:163065          Length:163065
## Class  :character      Class  :character
## Mode   :character      Mode   :character
##                               Mean    :47938
##                               3rd Qu.:72901
##                               Max.    :99835

```

3. Detección, tratamiento e imputación de datos faltantes

A través de la función summary empezamos comprobando que no hay datos faltantes en el data set. Por ello el grupo ha tenido que añadirlos manualmente para aproximarlos a un caso más real donde lo normal es encontrarlos y tener que lidiar con ellos. Los datos faltantes han sido imputados exclusivamente en las columnas que no van a servir de análisis principal para este estudio, para así intentar que la predicción que hagamos sea lo más precisa posible.

Utilizamos la librería missForest y generamos una semilla para que el resultado sea siempre el mismo.

```

set.seed(101)
hospital_charges <- bind_cols(hospital_charges[c(1, 3, 6, 7,
8, 11, 12)], missForest::prodNA(hospital_charges[c(-1, -3,
-6, -7, -8, -11, -12)], noNA = 0.1))

hospital_charges

## # A tibble: 163,065 x 12
##   'DRG Definition'      'Provider Name'     'Provider State' 'Provider Zip C~
##   <chr>                  <chr>                <chr>                  <dbl>
## 1 039 - EXTRACRANIAL PR~ SOUTHEAST ALABAMA M~ AL                    36301
## 2 039 - EXTRACRANIAL PR~ MARSHALL MEDICAL CE~ AL                    35957
## 3 039 - EXTRACRANIAL PR~ ELIZA COFFEE MEMORI~ AL                    35631
## 4 039 - EXTRACRANIAL PR~ ST VINCENT'S EAST AL                    35235
## 5 039 - EXTRACRANIAL PR~ SHELBY BAPTIST MEDI~ AL                    35007
## 6 039 - EXTRACRANIAL PR~ BAPTIST MEDICAL CEN~ AL                    36116
## 7 039 - EXTRACRANIAL PR~ EAST ALABAMA MEDICA~ AL                    36801
## 8 039 - EXTRACRANIAL PR~ UNIVERSITY OF ALABA~ AL                    35233
## 9 039 - EXTRACRANIAL PR~ HUNTSVILLE HOSPITAL AL                    35801

```

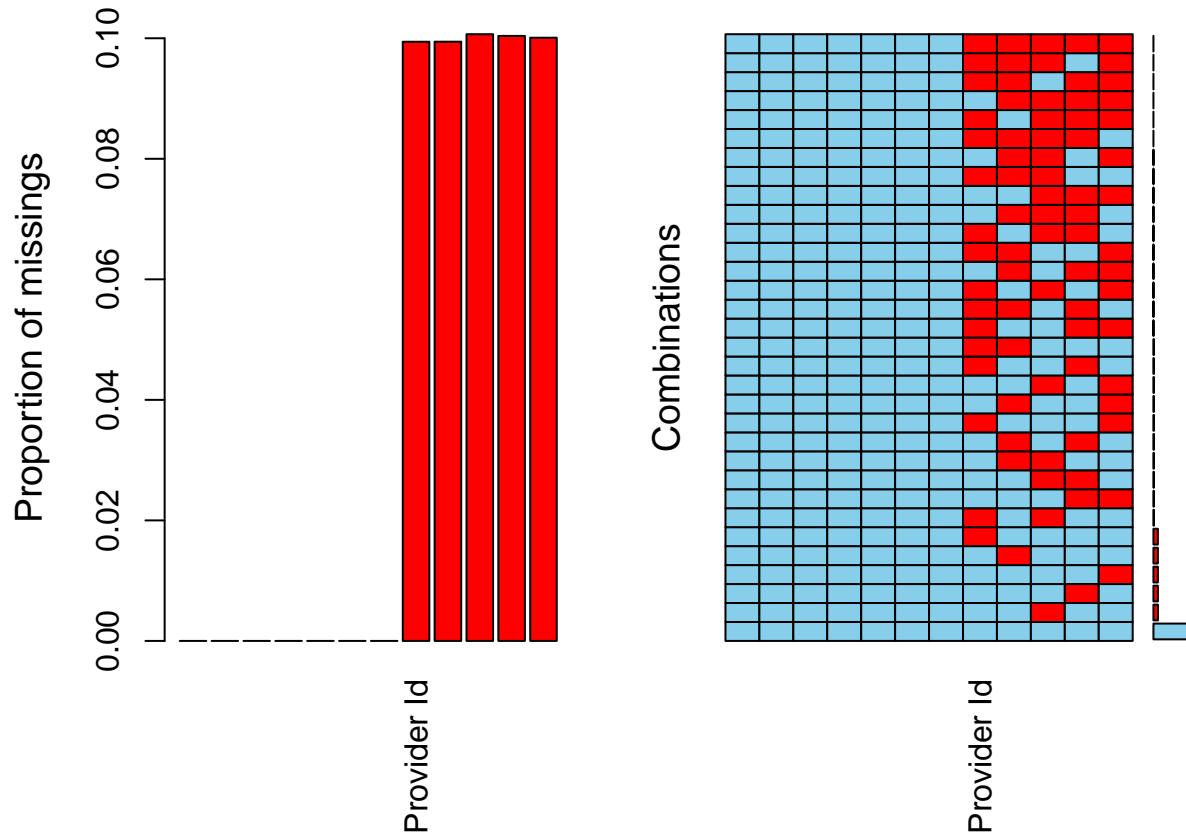
```

## 10 039 - EXTRACRANIAL PR~ GADSDEN REGIONAL ME~ AL          35903
## # ... with 163,055 more rows, and 8 more variables:
## #   Hospital Referral Region Description <chr>, Average Total Payments <chr>,
## #   Average Medicare Payments <chr>, Provider Id <dbl>,
## #   Provider Street Address <chr>, Provider City <chr>, Total Discharges <dbl>,
## #   Average Covered Charges <chr>

```

Haciendo uso de la librería VIM, analizamos un poco la estructura que tienen nuestros datos faltantes dentro de nuestra data set para ver y entender como se distribuyen.

```
summary(aggr(hospital_charges))
```



```

##
## Missings per variable:
##                               Variable Count
## DRG Definition           0
## Provider Name             0
## Provider State            0
## Provider Zip Code         0
## Hospital Referral Region Description  0
## Average Total Payments    0
## Average Medicare Payments 0
## Provider Id 16212
## Provider Street Address 16213
## Provider City 16416

```

```

##                               Total Discharges 16370
##                         Average Covered Charges 16321
##
##   ##  Missing values in combinations of variables:
##           Combinations Count      Percent
##   ##  0:0:0:0:0:0:0:0:0:0:0 96329 5.907399e+01
##   ##  0:0:0:0:0:0:0:0:0:0:1 10674 6.545856e+00
##   ##  0:0:0:0:0:0:0:0:0:1:0 10681 6.550149e+00
##   ##  0:0:0:0:0:0:0:0:0:1:1 1232 7.555269e-01
##   ##  0:0:0:0:0:0:0:0:1:0:0 10753 6.594303e+00
##   ##  0:0:0:0:0:0:0:0:1:0:1 1185 7.267041e-01
##   ##  0:0:0:0:0:0:0:0:1:1:0 1227 7.524607e-01
##   ##  0:0:0:0:0:0:0:0:1:1:1 124 7.604330e-02
##   ##  0:0:0:0:0:0:0:0:1:0:0 10651 6.531751e+00
##   ##  0:0:0:0:0:0:0:0:1:0:1 1192 7.309968e-01
##   ##  0:0:0:0:0:0:0:0:1:1:0 1209 7.414221e-01
##   ##  0:0:0:0:0:0:0:0:1:0:1 137 8.401558e-02
##   ##  0:0:0:0:0:0:0:0:1:1:0 1211 7.426486e-01
##   ##  0:0:0:0:0:0:0:0:1:1:1 108 6.623126e-02
##   ##  0:0:0:0:0:0:0:0:1:1:1:0 127 7.788305e-02
##   ##  0:0:0:0:0:0:0:0:1:1:1:1 13 7.972281e-03
##   ##  0:0:0:0:0:0:0:0:1:0:0:0 10626 6.516420e+00
##   ##  0:0:0:0:0:0:0:0:1:0:0:1 1200 7.359029e-01
##   ##  0:0:0:0:0:0:0:0:1:0:0:1:0 1156 7.089198e-01
##   ##  0:0:0:0:0:0:0:0:1:0:0:1:1 146 8.953485e-02
##   ##  0:0:0:0:0:0:0:0:1:0:1:0:0 1236 7.579799e-01
##   ##  0:0:0:0:0:0:0:0:1:0:1:0:1 140 8.585533e-02
##   ##  0:0:0:0:0:0:0:0:1:0:1:1:0 129 7.910956e-02
##   ##  0:0:0:0:0:0:0:0:1:0:1:1:1 14 8.585533e-03
##   ##  0:0:0:0:0:0:0:0:1:1:0:0:0 1127 6.911354e-01
##   ##  0:0:0:0:0:0:0:0:1:1:0:0:1 133 8.156257e-02
##   ##  0:0:0:0:0:0:0:0:1:1:0:0:1:0 144 8.830834e-02
##   ##  0:0:0:0:0:0:0:0:1:1:0:0:1:1 12 7.359029e-03
##   ##  0:0:0:0:0:0:0:0:1:1:1:0:0:0 120 7.359029e-02
##   ##  0:0:0:0:0:0:0:0:1:1:1:0:0:1 10 6.132524e-03
##   ##  0:0:0:0:0:0:0:0:1:1:1:1:0:0 18 1.103854e-02
##   ##  0:0:0:0:0:0:0:0:1:1:1:1:1:0:1 1 6.132524e-04

```

```
# referencia https://rpubs.com/sediaz/na\_aggr
```

4. Partición del conjunto de datos: data set training y data set test

Una vez vistos por encima la estructura general de los datos, y habiendo añadido los datos faltantes que nos hacían falta, pasamos a dividir el conjunto de datos en dos, para diferenciar los que usaremos de entrenamiento de los que usaremos de test (viendo la cantidad de datos de la que disponemos, la distribución elegida ha sido: 20% test y 80% training). Establecemos una semilla que nos guarde de forma permanente la división que hacemos, para que la división de los datos sea siempre la misma.

Guardamos además la partición de datos de test para ser utilizada a futuro para la validación del modelo final, y pasamos a trabajar de aquí en adelante con la partición de training.

```

set.seed(101)
sample <- sample.int(n = nrow(hospital_charges), size = floor(0.8 *
  nrow(hospital_charges)), replace = F)
train <- hospital_charges[sample, ]
test <- hospital_charges[-sample, ]

train

## # A tibble: 130,452 x 12
##   'DRG Definition'      'Provider Name'    'Provider State' 'Provider Zip C-
##   <chr>                  <chr>                <chr>           <dbl>
## 1 064 - INTRACRANIAL HEMO~ RIVERVIEW HOSPITAL IN          46060
## 2 439 - DISORDERS OF PAN~ ST LUKE'S ROOSEVE~ NY          10025
## 3 853 - INFECTIOUS & PARA~ ST JOSEPH'S MEDIC~ NY          10701
## 4 329 - MAJOR SMALL & LAR~ UNIVERSITY OF KAN~ KS          66160
## 5 195 - SIMPLE PNEUMONIA ~ GARDEN CITY HOSPI~ MI          48135
## 6 176 - PULMONARY EMBOLIS~ HORIZON MEDICAL C~ TN          37055
## 7 641 - MISC DISORDERS OF~ BAYLOR UNIVERSITY~ TX          75246
## 8 638 - DIABETES W CC     ST ELIZABETH FLOR~ KY          41042
## 9 872 - SEPTICEMIA OR SEV~ ST JOSEPH'S HOSPI~ NY          13203
## 10 439 - DISORDERS OF PAN~ SOUTH POINTE HOSP~ OH          44122
## # ... with 130,442 more rows, and 8 more variables:
## #   Hospital Referral Region Description <chr>, Average Total Payments <chr>,
## #   Average Medicare Payments <chr>, Provider Id <dbl>,
## #   Provider Street Address <chr>, Provider City <chr>, Total Discharges <dbl>,
## #   Average Covered Charges <chr>

test

## # A tibble: 32,613 x 12
##   'DRG Definition'      'Provider Name'    'Provider State' 'Provider Zip C-
##   <chr>                  <chr>                <chr>           <dbl>
## 1 039 - EXTRACRANIAL P~ MARSHALL MEDICAL CEN~ AL          35957
## 2 039 - EXTRACRANIAL P~ SOUTH BALDWIN REGION~ AL          36535
## 3 039 - EXTRACRANIAL P~ MOBILE INFIRMARY        AL          36652
## 4 039 - EXTRACRANIAL P~ TUCSON MEDICAL CENTER AZ          85712
## 5 039 - EXTRACRANIAL P~ CARONDELET ST JOSEPH~ AZ          85711
## 6 039 - EXTRACRANIAL P~ ST JOSEPH'S HOSPITAL~ AZ          85013
## 7 039 - EXTRACRANIAL P~ BANNER BOSWELL MEDIC~ AZ          85351
## 8 039 - EXTRACRANIAL P~ SUMMIT HEALTHCARE RE~ AZ          85901
## 9 039 - EXTRACRANIAL P~ BANNER HEART HOSPITAL AZ          85206
## 10 039 - EXTRACRANIAL P~ CONWAY REGIONAL MEDI~ AR         72034
## # ... with 32,603 more rows, and 8 more variables:
## #   Hospital Referral Region Description <chr>, Average Total Payments <chr>,
## #   Average Medicare Payments <chr>, Provider Id <dbl>,
## #   Provider Street Address <chr>, Provider City <chr>, Total Discharges <dbl>,
## #   Average Covered Charges <chr>

```

5. EDA - Análisis exploratorio de datos

5.1 Definición de las variables que componen los datos de estudio

Empezando ya el análisis en profundidad del conjunto de datos que tenemos, vemos que las 12 variables que componen los datos pueden ser descritas como:

- **DRG Definition:** Grupo relativo a un diagnóstico. Los grupos de diagnóstico relacionado (DRG) se utilizan para clasificar la gravedad de la enfermedad en las visitas hospitalarias de pacientes hospitalizados, el riesgo de mortalidad, el pronóstico, la dificultad del tratamiento, la necesidad de intervención y la intensidad de los recursos que necesitan. El sistema DRG fue desarrollado en la Universidad de Yale en la década de 1970 para la clasificación estadística de casos hospitalarios. Realmente la variable DRG es relativa al código y la descripción que identifican el MS-DRG. Los MS-DRG son un sistema de clasificación que agrupa condiciones clínicas similares (diagnósticos) y los procedimientos proporcionados por el hospital durante la estancia. El sistema de Medicare (Sistema de Seguridad Social en EEUU) los utiliza para determinar los reembolsos para hospitales, centros de enfermería especializada y hospicios. Una estadía en el hospital puede variar de un día a 100 días. Los MS-DRG más caros tienen las estadías promedio más largas. El establecimiento del cada DRG se establece según las condiciones clínicas del paciente, necesidad de cantidades similares de recursos para pacientes hospitalizados y sexo y edad del paciente. Para ello se utiliza el sistema de DRG llamado “Medicare Severity DRGs (MS-DRGs)” para reflejar en mejor manera la severidad de la enfermedad del paciente y su consumo de recursos para su recuperación. Para clasificar el nivel de severidad de un paciente dentro del sistema “MS-DRGs” hay códigos secundarios de diagnóstico:

- MCC: Major Complication/Comorbidity -> El nivel más alto de severidad.
- CC: Complication/Comorbidity -> El siguiente nivel de severidad.
- Non-CC: Non-Complication/Comorbidity -> Este nivel no supone una gran severidad en la enfermedad ni un gran gasto de recursos;
- **Provider ID:** ID o número identificativo de referencia del hospital;
- **Provider Name:** Nombre del hospital;
- **Provider Street Address:** Dirección postal donde se ubica el hospital;
- **Provider City:** Ciudad donde se ubica el hospital;
- **Provider State:** Estado federal de EEUU donde se ubica el hospital;
- **Provider Zip Code:** Código postal donde se ubica el hospital;
- **Hospital Referral Region Description:** Delinación geográfica específica creada por la organización norteamericana “Dartmouth Atlas of Health Care”, para estudiar los mercados vinculados al sector salud en EEUU;
- **Total Discharges:** Número de personas dadas de alta;
- **Average Covered Charges:** Gastos medios del hospital por los servicios cubiertos por la seguridad social para todas las altas del grupo relacionado con el diagnóstico. Por lo tanto cargo promedio según grupo de diagnóstico DRG establecido. Los pacientes que tienen características clínicas similares y costos de tratamiento similares se asignan a un Grupo de Diagnóstico Relacionado (DRG). El DRG está vinculado a un monto de pago fijo basado en el costo promedio del tratamiento de los pacientes del grupo. La asignación de DRG se basa en el diagnóstico del paciente, los procedimientos recibidos, la edad y otra información. Por lo tanto esta variable contiene el cargo promedio por cada DRG proporcionado por el hospital. Sus cargos promedio podrían ser más o menos dependiendo de las necesidades específicas de su paciente y los servicios prestados. Esto es lo que el hospital cobra en la factura final del hospital y es equivalente al “sticker price”. Este es en gran medida un número

irrelevante, ya que no importa lo que cobren los diferentes hospitales, a todos se les pagará la misma cantidad de Medicare por cualquier DRG dado. Prácticamente nadie paga el “stiker price” en un hospital. Cuando un paciente ha sido admitido como hospitalizado en un hospital, ese hospital asigna un DRG cuando este paciente es dado de alta, basándolo en la atención que necesitaba durante su estadía en el hospital. Al hospital se le paga una cantidad fija por ese DRG, independientemente de cuánto dinero realmente gaste en su tratamiento. Si un hospital puede tratar a un paciente de forma efectiva por menos dinero del que Medicare paga por su DRG, entonces el hospital gana dinero con esa hospitalización. Si el hospital gasta más dinero cuidando del paciente de lo que Medicare le da para su DRG, entonces el hospital pierde dinero en esa hospitalización;

- **Average Medicare Payments:** Importe medio cubierto por la Seguridad Social de EEUU. Esto es lo que Medicare paga al hospital por ese DRG;
- **Average Total Payments:** Importe medio total a pagar por persona. Esto es lo que realmente se le paga al hospital e incluye lo que paga Medicare más los copagos que paga el paciente más cualquier cosa que pague el seguro secundario (seguro privado).

5.2. Definición de objetivos

El objetivo final del proyecto es llegar a un modelo que permita recomendar cual es el hospital o grupo de hospitales óptimo que debe elegir un paciente enfermo en EEUU, en base a la posible enfermedad que le van a diagnosticar, su localización geográfica y los costes que su caso clínico puede llegar a tener en base al sistema sanitario estadounidense.

Para esta primera entrega, el objetivo es realizar el tratamiento de datos adecuado y seleccionar las mejores variables que servirán para llegar al modelo de Machine Learning deseado. Se realizará de la misma manera un ajuste, interpretación y diagnosis del modelo de regresión lineal múltiple, en base a las variables que mejor expliquen los datos.

5.3. Transformaciones de variables cuantitativas y procesado de variables cualitativas - Limpieza de datos

5.3.1 Cambiar los nombres de las columnas

Se ha decidido realizar un cambio en el nombre de las variables que aparecen en las columnas de los datos para así seguir un mismo patrón y a al vez evitar tener espacios que nos pueden llegar a dar problemas a futuro.

```
train
```

```
## # A tibble: 130,452 x 12
##   `DRG Definition` `Provider Name` `Provider State` `Provider Zip C~
##   <chr>            <chr>           <chr>           <dbl>
## 1 064 - INTRACRANIAL HEMO~ RIVERVIEW HOSPITAL IN      46060
## 2 439 - DISORDERS OF PAN~ ST LUKE'S ROOSEVE~ NY      10025
## 3 853 - INFECTIOUS & PARA~ ST JOSEPH'S MEDIC~ NY      10701
## 4 329 - MAJOR SMALL & LAR~ UNIVERSITY OF KAN~ KS      66160
## 5 195 - SIMPLE PNEUMONIA ~ GARDEN CITY HOSPI~ MI      48135
## 6 176 - PULMONARY EMBOLIS~ HORIZON MEDICAL C~ TN      37055
## 7 641 - MISC DISORDERS OF~ BAYLOR UNIVERSITY~ TX      75246
## 8 638 - DIABETES W CC    ST ELIZABETH FLOR~ KY      41042
```

```

## 9 872 - SEPTICEMIA OR SEV~ ST JOSEPH'S HOSPI~ NY 13203
## 10 439 - DISORDERS OF PAN~ SOUTH POINTE HOSP~ OH 44122
## # ... with 130,442 more rows, and 8 more variables:
## #   Hospital Referral Region Description <chr>, Average Total Payments <chr>,
## #   Average Medicare Payments <chr>, Provider Id <dbl>,
## #   Provider Street Address <chr>, Provider City <chr>, Total Discharges <dbl>,
## #   Average Covered Charges <chr>

names(train) <- c("drg_def", "prov_name", "prov_state", "prov_zip",
  "referral_reg", "mean_total_payments", "mean_medicare_payments",
  "prov_id", "prov_address", "prov_city", "total_discharges",
  "mean_covered_charges")

head(train)

## # A tibble: 6 x 12
##   drg_def      prov_name    prov_state prov_zip referral_reg mean_total_paym-
##   <chr>        <chr>       <chr>      <dbl> <chr>          <chr>
## 1 064 - INTRACRA~ RIVERVIEW H~ IN           46060 IN - Indian~ $11085.90
## 2 439 - DISORDER~ ST LUKE'S R~ NY          10025 NY - Manhat~ $11852.61
## 3 853 - INFECTIO~ ST JOSEPH'S~ NY         10701 NY - White ~ $47649.00
## 4 329 - MAJOR SM~ UNIVERSITY ~ KS          66160 MO - Kansas~ $42983.75
## 5 195 - SIMPLE P~ GARDEN CITY~ MI         48135 MI - Dearbo~ $5235.28
## 6 176 - PULMONAR~ HORIZON MED~ TN         37055 TN - Nashvi~ $6612.35
## # ... with 6 more variables: mean_medicare_payments <chr>, prov_id <dbl>,
## #   prov_address <chr>, prov_city <chr>, total_discharges <dbl>,
## #   mean_covered_charges <chr>

```

5.3.2 División de la columna drg_def

Realizamos una división de la columna “drg_ref”. Separamos la columna en dos diferenciando entre código de la enfermedad y descripción de la enfermedad. Nos servirá a futuro para simplificar el análisis y visualización de los datos de interés.

```

train <- train %>%
  separate(data = ., col = drg_def,
  into = c("codigo_enf", "desc_enf"), sep = "-")
train

## # A tibble: 130,452 x 13
##   codigo_enf desc_enf      prov_name    prov_state prov_zip referral_reg
##   <chr>      <chr>       <chr>       <chr>      <dbl> <chr>
## 1 "064 "     " INTRACRANIAL HEM~ RIVERVIEW HO~ IN           46060 IN - Indian-
## 2 "439 "     " DISORDERS OF PAN~ ST LUKE'S RO~ NY          10025 NY - Manhat-
## 3 "853 "     " INFECTIOUS & PAR~ ST JOSEPH'S ~ NY         10701 NY - White ~
## 4 "329 "     " MAJOR SMALL & LA~ UNIVERSITY O~ KS          66160 MO - Kansas~
## 5 "195 "     " SIMPLE PNEUMONIA~ GARDEN CITY ~ MI         48135 MI - Dearbo~
## 6 "176 "     " PULMONARY EMBOLI~ HORIZON MEDI~ TN         37055 TN - Nashvi~
## 7 "641 "     " MISC DISORDERS O~ BAYLOR UNIVE~ TX         75246 TX - Dallas
## 8 "638 "     " DIABETES W CC"  ST ELIZABETH~ KY          41042 KY - Coving-
## 9 "872 "     " SEPTICEMIA OR SE~ ST JOSEPH'S ~ NY         13203 NY - Syracu-
## 10 "439 "    " DISORDERS OF PAN~ SOUTH POINTE~ OH        44122 OH - Clevel-

```

```

## # ... with 130,442 more rows, and 7 more variables: mean_total_payments <chr>,
## #   mean_medicare_payments <chr>, prov_id <dbl>, prov_address <chr>,
## #   prov_city <chr>, total_discharges <dbl>, mean_covered_charges <chr>

```

5.3.3 Cambio de tipo de variable

Se ha decidido eliminar el símbolo de moneda de dólar de las últimas tres columnas, transformando las columnas a tipo numérico.

```

train$mean_covered_charges = as.numeric(gsub("\\\\$", "", train$mean_covered_charges))

train$mean_total_payments = as.numeric(gsub("\\\\$", "", train$mean_total_payments))

train$mean_medicare_payments = as.numeric(gsub("\\\\$", "", train$mean_medicare_payments))

train$prov_zip = as.factor(train$prov_zip)

train$prov_id = as.factor(train$prov_id)

head(train)

## # A tibble: 6 x 13
##   codigo_enf desc_enf      prov_name    prov_state prov_zip referral_reg
##   <chr>     <chr>      <chr>        <chr>       <fct>      <chr>
## 1 "064 "    " INTRACRANIAL HEM~ RIVERVIEW HO~ IN      46060  IN - Indiana-
## 2 "439 "    " DISORDERS OF PAN~ ST LUKE'S RO~ NY      10025  NY - Manhatt-
## 3 "853 "    " INFECTIOUS & PAR~ ST JOSEPH'S ~ NY     10701  NY - White P-
## 4 "329 "    " MAJOR SMALL & LA~ UNIVERSITY O~ KS      66160  MO - Kansas ~
## 5 "195 "    " SIMPLE PNEUMONIA~ GARDEN CITY ~ MI     48135  MI - Dearborn
## 6 "176 "    " PULMONARY EMBOLI~ HORIZON MEDI~ TN      37055  TN - Nashvil-
## # ... with 7 more variables: mean_total_payments <dbl>,
## #   mean_medicare_payments <dbl>, prov_id <fct>, prov_address <chr>,
## #   prov_city <chr>, total_discharges <dbl>, mean_covered_charges <dbl>

str(train)

## tibble [130,452 x 13] (S3: tbl_df/tbl/data.frame)
## $ codigo_enf      : chr [1:130452] "064 " "439 " "853 " "329 " ...
## $ desc_enf        : chr [1:130452] " INTRACRANIAL HEMORRHAGE OR CEREBRAL INFARCTION W MCC" " ...
## $ prov_name       : chr [1:130452] "RIVERVIEW HOSPITAL" "ST LUKE'S ROOSEVELT HOSPITAL" "ST JOSEPH'S ...
## $ prov_state      : chr [1:130452] "IN" "NY" "NY" "KS" ...
## $ prov_zip        : Factor w/ 3040 levels "1040","1060",...: 1471 203 221 1991 1557 1132 2271 ...
## $ referral_reg    : chr [1:130452] "IN - Indianapolis" "NY - Manhattan" "NY - White Plains" "NY - ...
## $ mean_total_payments : num [1:130452] 11086 11853 47649 42984 5235 ...
## $ mean_medicare_payments: num [1:130452] 8772 11076 44184 41458 4358 ...
## $ prov_id         : Factor w/ 3309 levels "10001","10005",...: 1056 NA 1915 NA 1551 2668 NA NA ...
## $ prov_address    : chr [1:130452] "395 WESTFIELD RD" "1111 AMSTERDAM AVENUE" "127 SOUTH BROADWAY" ...
## $ prov_city        : chr [1:130452] "NOBLESVILLE" "NEW YORK" "YONKERS" "KANSAS CITY" ...
## $ total_discharges : num [1:130452] NA 13 17 NA 52 14 110 15 70 NA ...
## $ mean_covered_charges : num [1:130452] 42249 40094 NA 187656 9830 ...

```

5.3.4 Creando columna nueva relativa a los copagos que deben realizar los pacientes: mean_total_payments - mean_medicare_payments

Nueva variable representativa del valor de los copagos que debe realizar el paciente o su seguro privado (en caso de contar con uno), para completar, junto a lo que cubre el Estado con el Medicare, el coste total de la intervención hospitalaria.

```
train <- train %>%
  mutate(copagos = mean_total_payments - mean_medicare_payments)
train

## # A tibble: 130,452 x 14
##   codigo_enf desc_enf      prov_name prov_state prov_zip referral_reg
##   <chr>       <chr>       <chr>       <chr>       <fct>     <chr>
## 1 "064 "     " INTRACRANIAL HEM~ RIVERVIEW HO~ IN    46060  IN - Indian-
## 2 "439 "     " DISORDERS OF PAN~ ST LUKE'S RO~ NY   10025  NY - Manhat-
## 3 "853 "     " INFECTIOUS & PAR~ ST JOSEPH'S ~ NY  10701  NY - White ~
## 4 "329 "     " MAJOR SMALL & LA~ UNIVERSITY O~ KS  66160  MO - Kansas-
## 5 "195 "     " SIMPLE PNEUMONIA~ GARDEN CITY ~ MI  48135  MI - Dearbo~
## 6 "176 "     " PULMONARY EMBOLI~ HORIZON MEDI~ TN  37055  TN - Nashvi~
## 7 "641 "     " MISC DISORDERS O~ BAYLOR UNIVE~ TX  75246  TX - Dallas
## 8 "638 "     " DIABETES W CC"  ST ELIZABETH~ KY  41042  KY - Coving~
## 9 "872 "     " SEPTICEMIA OR SE~ ST JOSEPH'S ~ NY  13203  NY - Syracu~
## 10 "439 "    " DISORDERS OF PAN~ SOUTH POINTE~ OH  44122  OH - Clevel-
## # ... with 130,442 more rows, and 8 more variables: mean_total_payments <dbl>,
## #   mean_medicare_payments <dbl>, prov_id <fct>, prov_address <chr>,
## #   prov_city <chr>, total_discharges <dbl>, mean_covered_charges <dbl>,
## #   copagos <dbl>
```

5.3.5 Creando columna nueva relativa a la tasa de cobertura de la Seguridad Social estadounidense: mean_medicare_payments/mean_total_payments

Nueva variable representativa de la tasa de cobertura que ofrece el sistema de salud de la Seguridad Social americana según el hospital, el grupo de diagnóstico y la gravedad del paciente. Nos ayudamos para ello de la librería formattable, obteniendo resultados en forma de porcentaje de cobertura sobre el total a pagar al hospital.

```
train <- train %>% mutate(cobertura = percent(mean_medicare_payments/mean_total_payments))
train

## # A tibble: 130,452 x 15
##   codigo_enf desc_enf      prov_name prov_state prov_zip referral_reg
##   <chr>       <chr>       <chr>       <chr>       <fct>     <chr>
## 1 "064 "     " INTRACRANIAL HEM~ RIVERVIEW HO~ IN    46060  IN - Indian-
## 2 "439 "     " DISORDERS OF PAN~ ST LUKE'S RO~ NY   10025  NY - Manhat-
## 3 "853 "     " INFECTIOUS & PAR~ ST JOSEPH'S ~ NY  10701  NY - White ~
## 4 "329 "     " MAJOR SMALL & LA~ UNIVERSITY O~ KS  66160  MO - Kansas-
## 5 "195 "     " SIMPLE PNEUMONIA~ GARDEN CITY ~ MI  48135  MI - Dearbo~
## 6 "176 "     " PULMONARY EMBOLI~ HORIZON MEDI~ TN  37055  TN - Nashvi~
## 7 "641 "     " MISC DISORDERS O~ BAYLOR UNIVE~ TX  75246  TX - Dallas
## 8 "638 "     " DIABETES W CC"  ST ELIZABETH~ KY  41042  KY - Coving~
## 9 "872 "     " SEPTICEMIA OR SE~ ST JOSEPH'S ~ NY  13203  NY - Syracu~
```

```

## 10 "439 "      " DISORDERS OF PAN~ SOUTH POINTE~ OH          44122    OH - Clevel~
## # ... with 130,442 more rows, and 9 more variables: mean_total_payments <dbl>,
## #   mean_medicare_payments <dbl>, prov_id <fct>, prov_address <chr>,
## #   prov_city <chr>, total_discharges <dbl>, mean_covered_charges <dbl>,
## #   copagos <dbl>, cobertura <formtbl>

```

5.4 Análisis exhaustivo de los datos críticos para el estudio

5.4.1 Gráfico EEUU: valor de los copagos por Estados

```

# Haciendo la media de lo que cobra el hospital por estado

region_geog <- train %>%
  group_by(prov_state) %>%
  summarise(mean_total_price = mean(copagos))
region_geog

## # A tibble: 51 x 2
##       prov_state mean_total_price
##   <chr>           <dbl>
## 1 AK            1681.
## 2 AL            1148.
## 3 AR            1093.
## 4 AZ            1330.
## 5 CA            1140.
## 6 CO            1341.
## 7 CT            1266.
## 8 DC            1211.
## 9 DE            1389.
## 10 FL           1165.
## # ... with 41 more rows

# Libreria usmap tiene el mapa de EEUU por estado
library(usmap)

statepop #en libreria usmap hay un dataframe que es la poblacion para cada estado

## # A tibble: 51 x 4
##       fips abbr full          pop_2015
##   <chr> <chr> <chr>        <dbl>
## 1 01    AL    Alabama     4858979
## 2 02    AK    Alaska      738432
## 3 04    AZ    Arizona     6828065
## 4 05    AR    Arkansas    2978204
## 5 06    CA    California  39144818
## 6 08    CO    Colorado     5456574
## 7 09    CT    Connecticut 3590886
## 8 10    DE    Delaware     945934
## 9 11    DC    District of Columbia 672228
## 10 12   FL    Florida      20271272
## # ... with 41 more rows

```

```

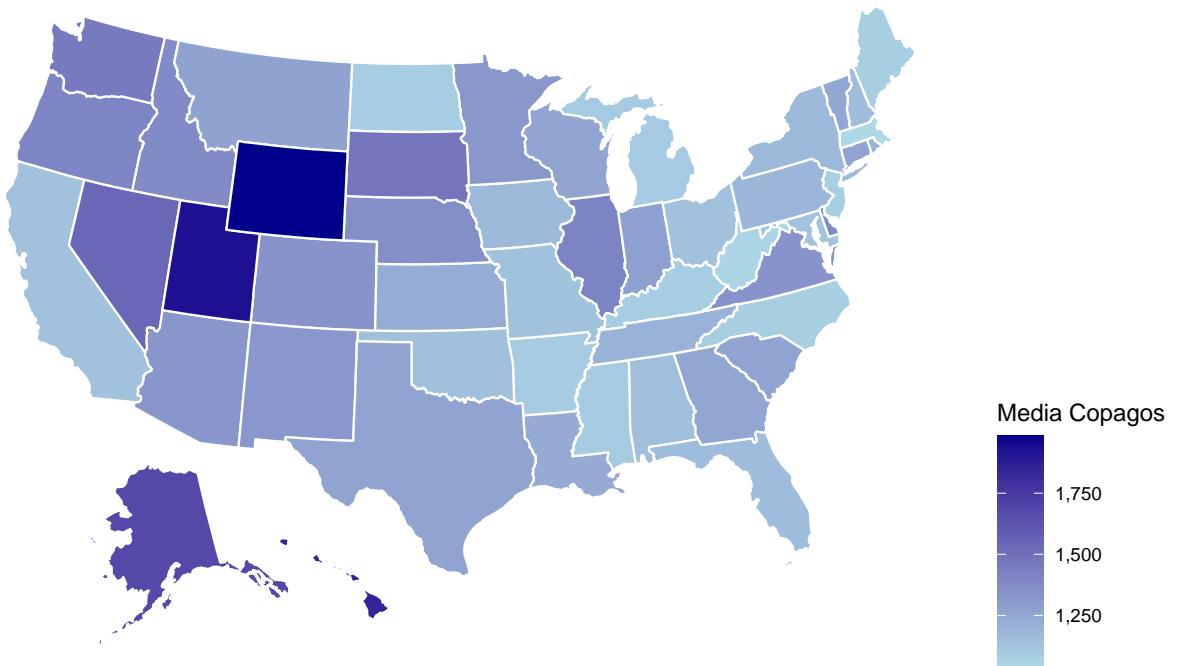
# (siglas -abbr ) y nos interesa agrupar a este data frame
# la columna mean_total_payments

names(statepop) <- c("fips", "prov_state", "full", "pop_2015") #cambiamos el nombre de la
# columna abbr para prov_state para tenerla igual en
# statepop y region_geog

statepop <- statepop %>%
  left_join(region_geog, by = "prov_state") #juntamos region_geog y statepop

plot_usmap(data = statepop, values = "mean_total_price", color = "white") +
  scale_fill_continuous(low = "light blue", high = "dark blue",
    name = "Media Copagos", label = scales::comma) + theme(legend.position = "right")

```



Los estados americanos más caros para el paciente son Wyoming (copagos medio de 1983 dólares), Utah (copagos medio de 1927 dólares) y Hawái (copagos medio de 1851 dólares)

Wyoming es el estado menos poblado de EEUU (dos tercios del territorio cubiertos por sierras y montañas), los componentes de su economía difieren significativamente de los otros estados (minería y turismo) y el gobierno es dueño de 50% de sus tierras.

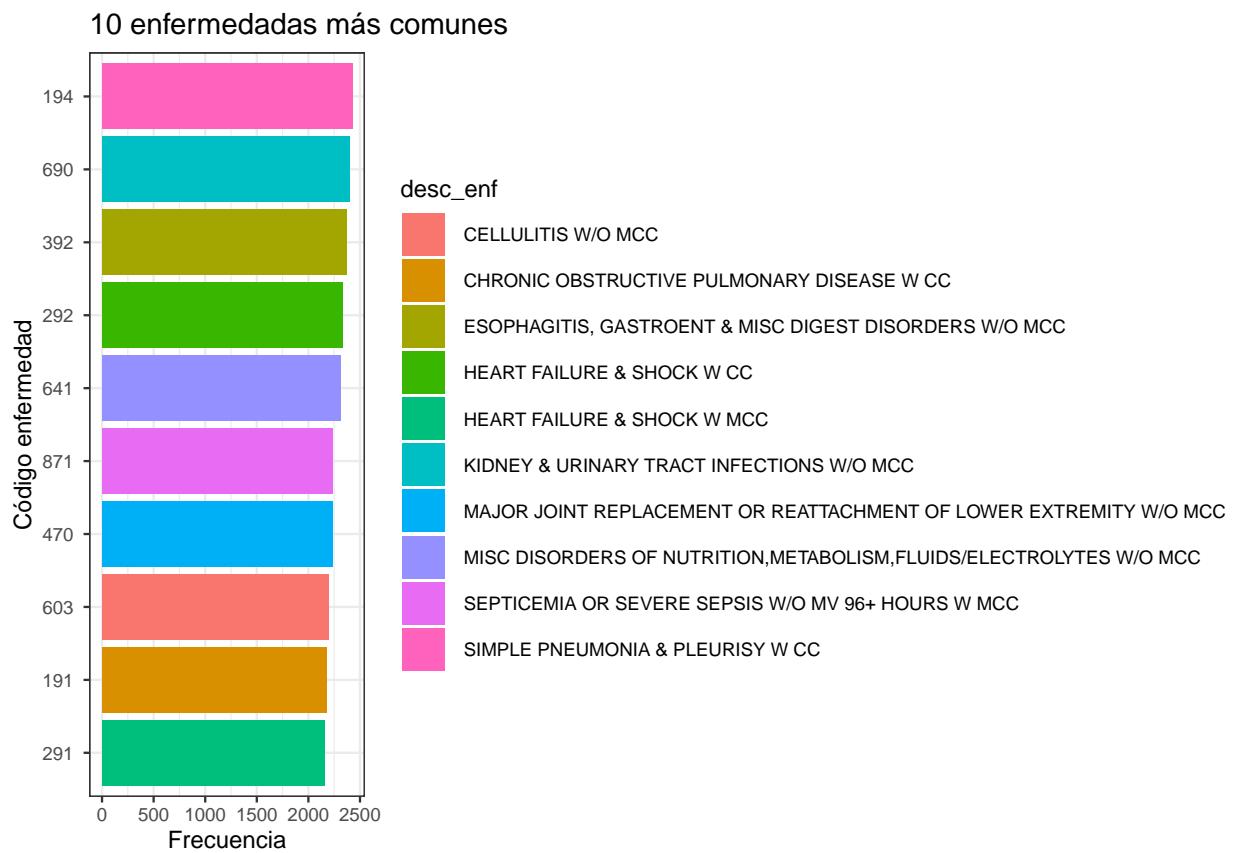
5.4.2 Top 10 enfermedades más comunes detectadas

```

d2 <- train %>%
  count(codigo_enf) %>%
  top_n(10) %>%
  arrange(n, codigo_enf) %>%
  mutate(codigo_enf = factor(codigo_enf, levels = unique(codigo_enf)))

train %>%
  filter(codigo_enf %in% d2$codigo_enf) %>%
  mutate(codigo_enf = factor(codigo_enf, levels = levels(d2$codigo_enf))) %>%
  ggplot(aes(x = codigo_enf, fill=desc_enf)) + geom_bar() + coord_flip() +
  theme_bw(base_size=9) + xlab("Código enfermedad") +
  ylab("Frecuencia") +
  ggtitle("10 enfermedades más comunes")

```



El grupo DRG de mayor frecuencia es “194 - Pneumonia y Pleuresía con complicaciones/comorbilidades” con 2.424 casos. En segundo y tercero lugar tenemos las “690 - Infecciones de las vías urinarias” (2.400 casos) y “392 -Esofagitis, Gastroenteritis y Enfermedades digestivas con” (2.368 casos) con o sin complicaciones.

5.4.3 Top 10 enfermedades más caras

```

d3 <- train %>%
  group_by(codigo_enf) %>% summarise(mean=mean(mean_total_payments)) %>% arrange(desc(mean))

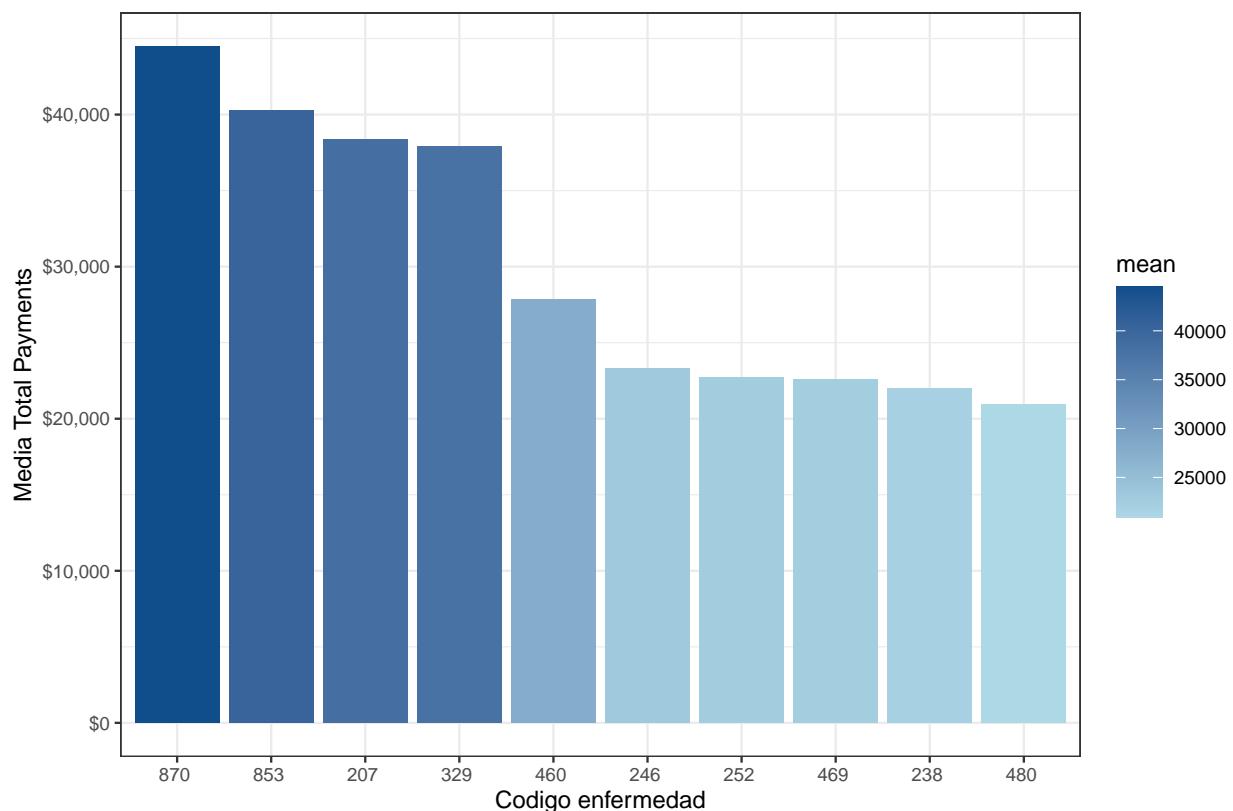
```

```
top_10_caras <- head(d3, 10)
top_10_caras
```

```
## # A tibble: 10 x 2
##   codigo_enf    mean
##   <chr>      <dbl>
## 1 "870 "     44467.
## 2 "853 "     40296.
## 3 "207 "     38372.
## 4 "329 "     37913.
## 5 "460 "     27822.
## 6 "246 "     23337.
## 7 "252 "     22731.
## 8 "469 "     22618.
## 9 "238 "     22009.
## 10 "480 "    20968.
```

```
ggplot(data=top_10_caras, mapping = aes(x = reorder(codigo_enf,-mean),mean)) + geom_bar(stat = "identity")
```

10 enfermedades más caras



El grupo DRG más caro es “870 - Septicemia y sepsis grave con periodo de hospitalización superior a 96 horas”. Una septicemia ocurre cuando el sistema inmunitario se descontrola y ataca a sus propios órganos y tejidos. Es la complicación de una infección, siendo una urgencia médica que requiere tratamiento inmediato. El coste media del tratamiento en los EEUU es de 44.467 dólares.

El segundo y tercero grupo DRG más caros son las “853 - Infecciones y enfermedades parasitarias” (coste

médio de 40.296 dólares) y "207- Enfermedades respiratorias que requieren ventilación respiratória por más de 96 horas consecutivas (coste medio de 38372.14 dólares).

5.4.4 Gráfico de calor - ratio de cobertura por Estado para las 10 enfermedades más caras

```
test <- top_10_caras %>% inner_join(train)
```

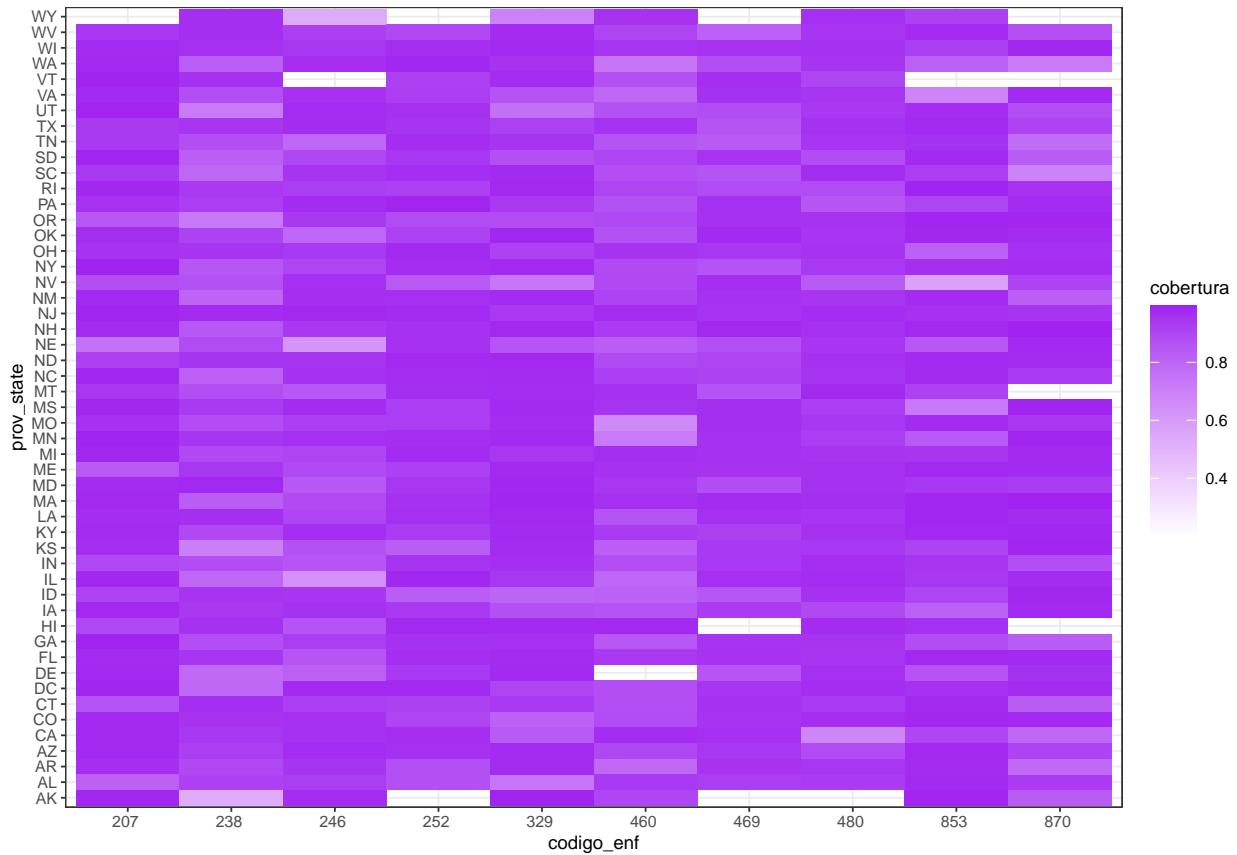
```
test
```



```
## # A tibble: 9,048 x 16
##   codigo_enf  mean_desc_enf prov_name  prov_state prov_zip referral_reg
##   <chr>        <dbl> <chr>       <chr>      <fct>    <chr>
## 1 "870 "     44467. " SEPTICEMIA~ SOUTHWEST G~ OH      44130  OH - Clevel~
## 2 "870 "     44467. " SEPTICEMIA~ DOWNEY REGI~ CA      90241  CA - Los An~
## 3 "870 "     44467. " SEPTICEMIA~ CORONA REGI~ CA      92882  CA - Orange~
## 4 "870 "     44467. " SEPTICEMIA~ ST JOSEPH'S~ NJ      7503   NJ - Paters~
## 5 "870 "     44467. " SEPTICEMIA~ UNIVERSITY ~ WA      98195   WA - Seattle
## 6 "870 "     44467. " SEPTICEMIA~ KINGSTON HO~ NY      12401   NY - Albany
## 7 "870 "     44467. " SEPTICEMIA~ MERCY HEALT~ MI      49444   MI - Muskeg~
## 8 "870 "     44467. " SEPTICEMIA~ ST LUKE'S H~ OH      43537   OH - Toledo
## 9 "870 "     44467. " SEPTICEMIA~ UNIVERSITY ~ AL      35233   AL - Birmin~
## 10 "870 "    44467. " SEPTICEMIA~ BETH ISRAEL~ MA      2215    MA - Boston
## # ... with 9,038 more rows, and 9 more variables: mean_total_payments <dbl>,
## #   mean_medicare_payments <dbl>, prov_id <fct>, prov_address <chr>,
## #   prov_city <chr>, total_discharges <dbl>, mean_covered_charges <dbl>,
## #   copagos <dbl>, cobertura <formtbl>
```



```
ggplot(test, aes(codigo_enf,prov_state, fill=cobertura))+geom_tile() + theme_bw(base_size=7) + scale_fi
```



De manera general se puede decir que la tasa de cobertura de las top 10 enfermedades más caras es, en todos los estados de Estados Unidos, superior a 80% (colores morado oscuro predominante en casi todo el gráfico).

La tasa de cobertura para la enfermedad Septicemia es en media, para todos los estados, de 95%, siendo entonces la enfermedad más cara para el sistema público, pero con gran cobertura y pocos copagos al paciente. Lo mismo se puede observar para las Enfermedades Parasitarias (94%) y Enfermedades Respiratoria (93%).

En el de Wyoming, el estado donde se hay que pagar las mayores cifras de copagos, no han sido registrados casos de Septicemia o Enfermedades Respiratorias, mientras la tasa de cobertura para Enfermedades Parasitarias es de 82%.

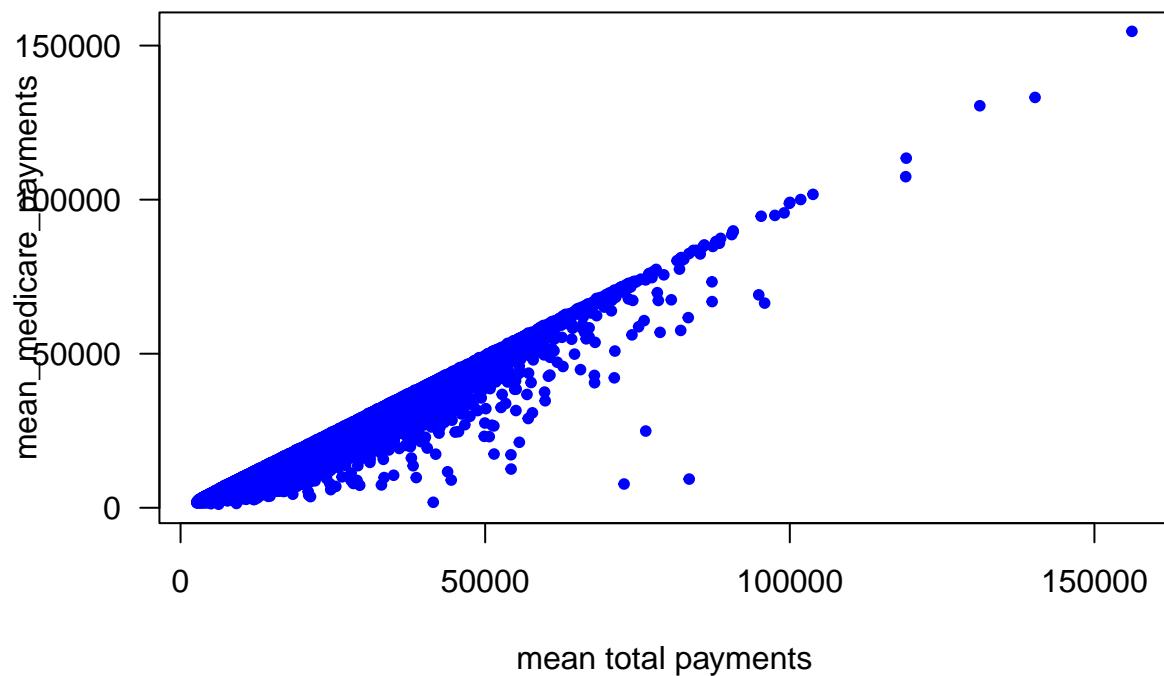
5.4.5 Correlación entre variables

```
library(PerformanceAnalytics)

cor(x=train$mean_total_payments, y=train$mean_medicare_payments)

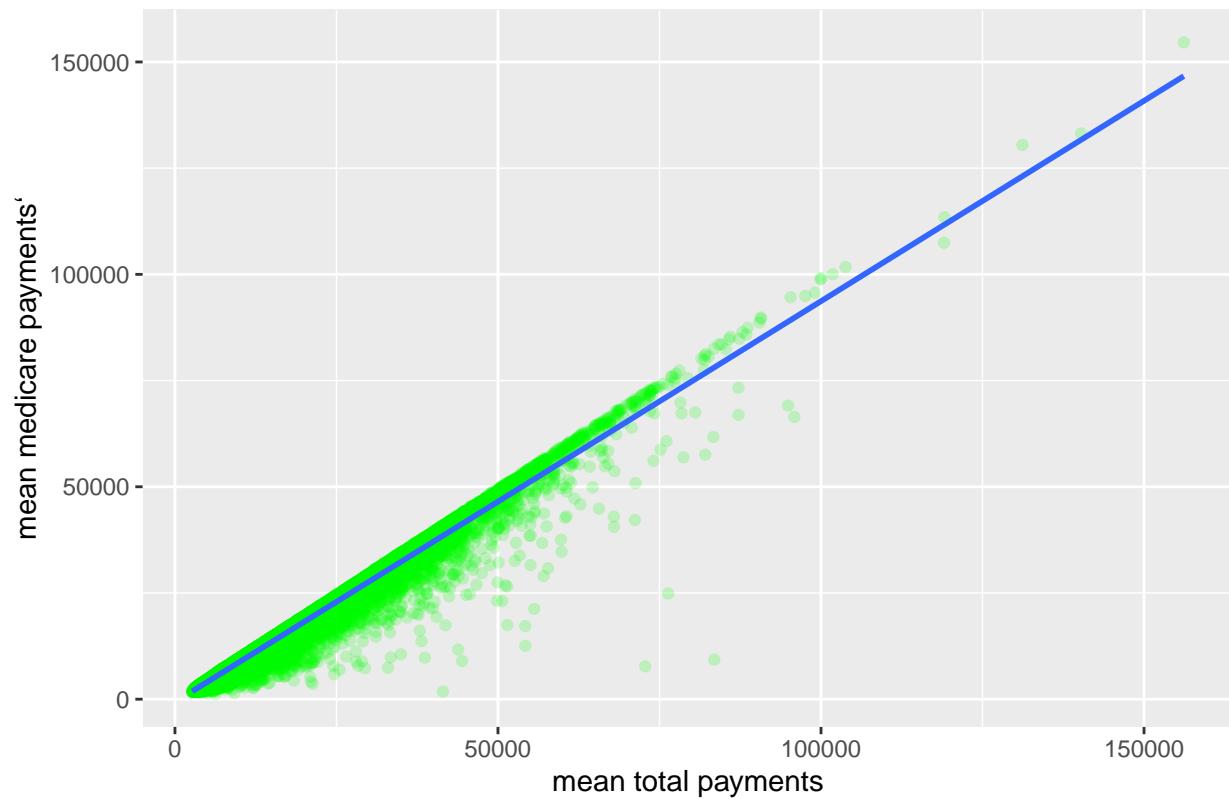
## [1] 0.9893899

with(train, plot(x=mean_total_payments, y=mean_medicare_payments, pch=20, col='blue',
                 xlab='mean total payments', las=1,
                 ylab='mean_medicare_payments'))
```



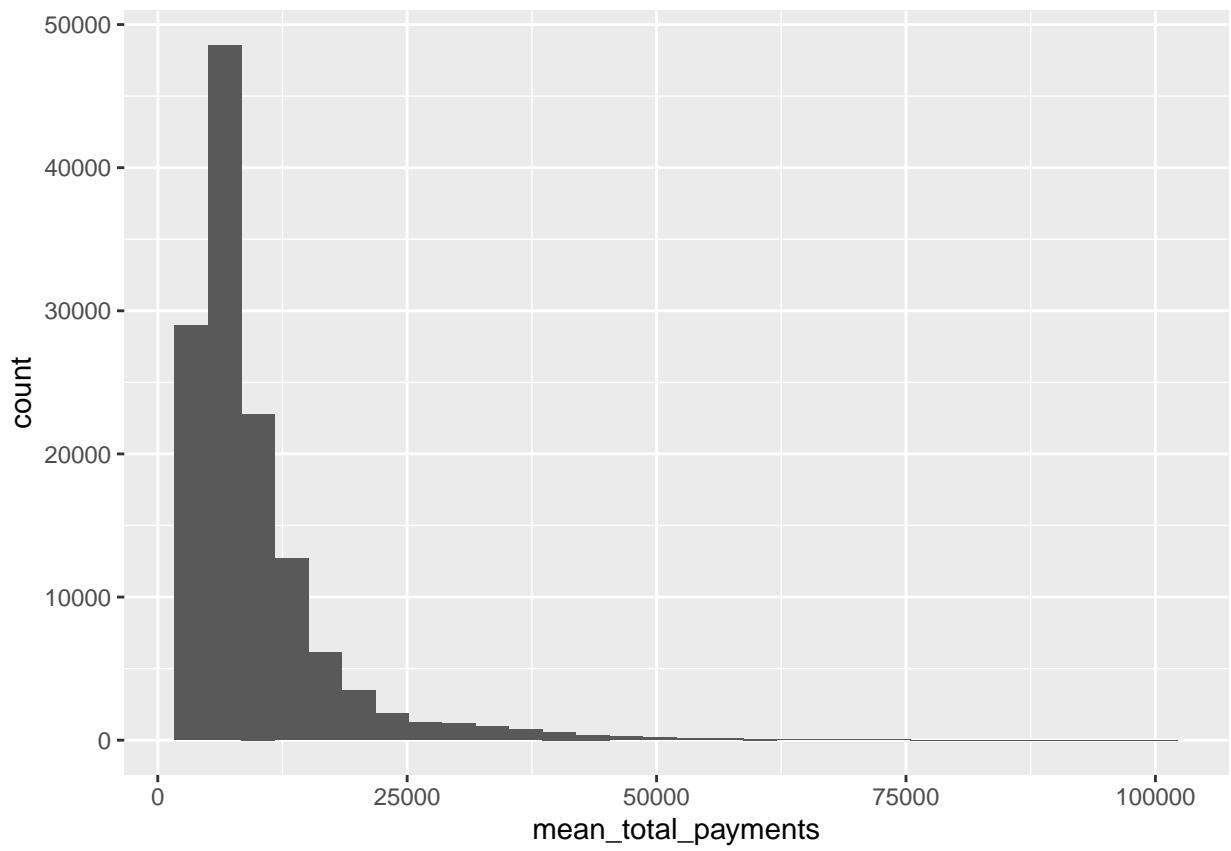
```
library(dplyr)
library(ggplot2)
train %>% ggplot(aes(mean_total_payments, mean_medicare_payments)) +
  geom_point(alpha=0.2, colour="green") +
  geom_smooth(formula= 'y ~ x',method = 'lm') +
  labs(title='Relacion entre variables total payments y medicare payments',
       x='mean total payments',
       y='mean medicare payments')
```

Relacion entre variables total payments y medicare payments

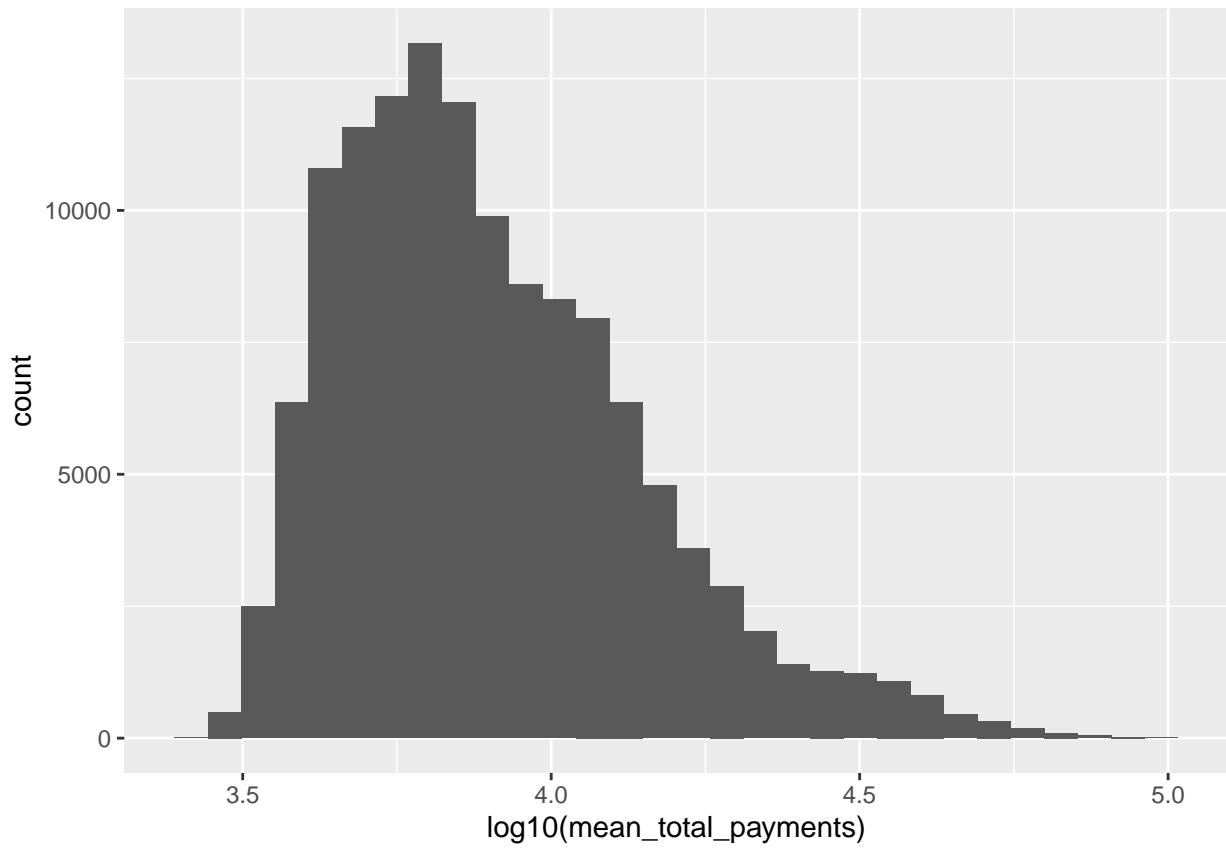


5.4.6 Análisis de la distribución de las variables

```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_total_payments<100000 ) %>%
  ggplot(aes(x=mean_total_payments))+ geom_histogram()
```

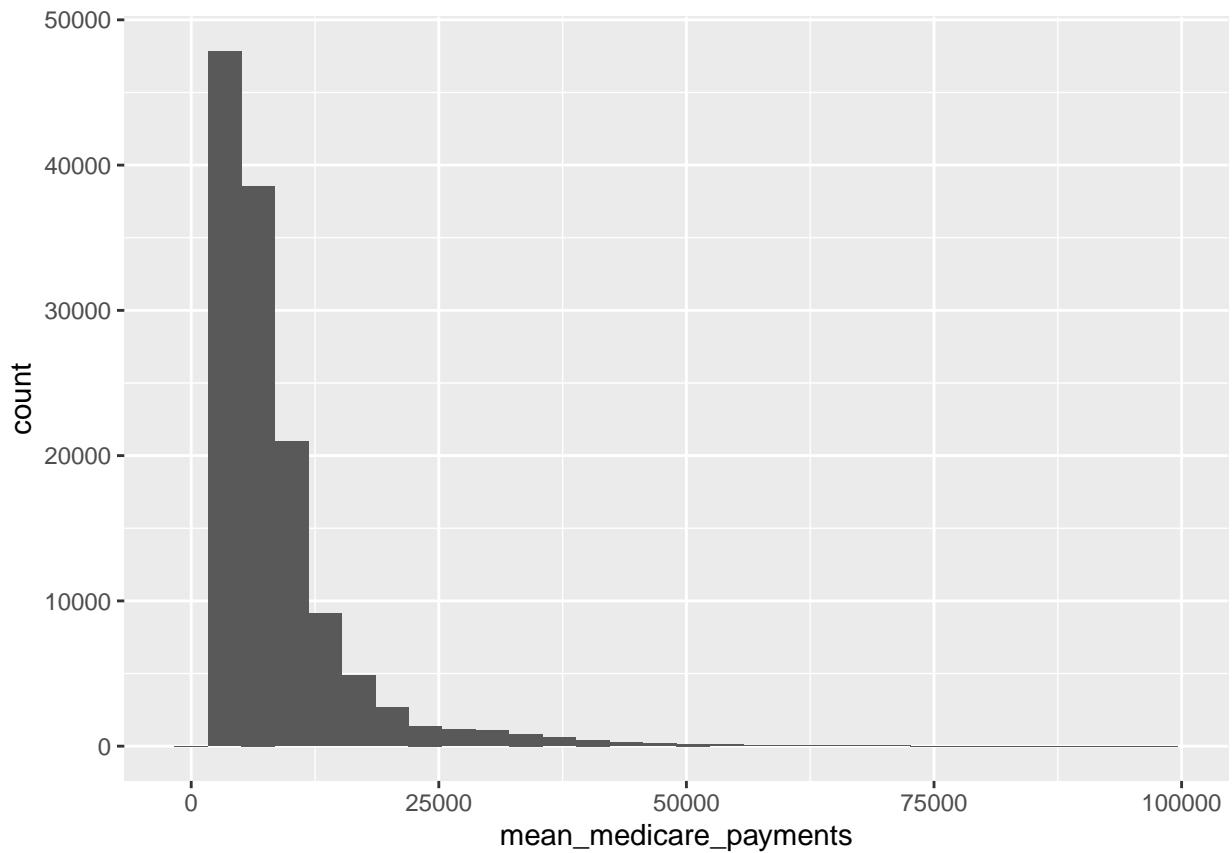


```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_total_payments<100000 ) %>%
  ggplot(aes(x=log10(mean_total_payments)))+ geom_histogram()
```

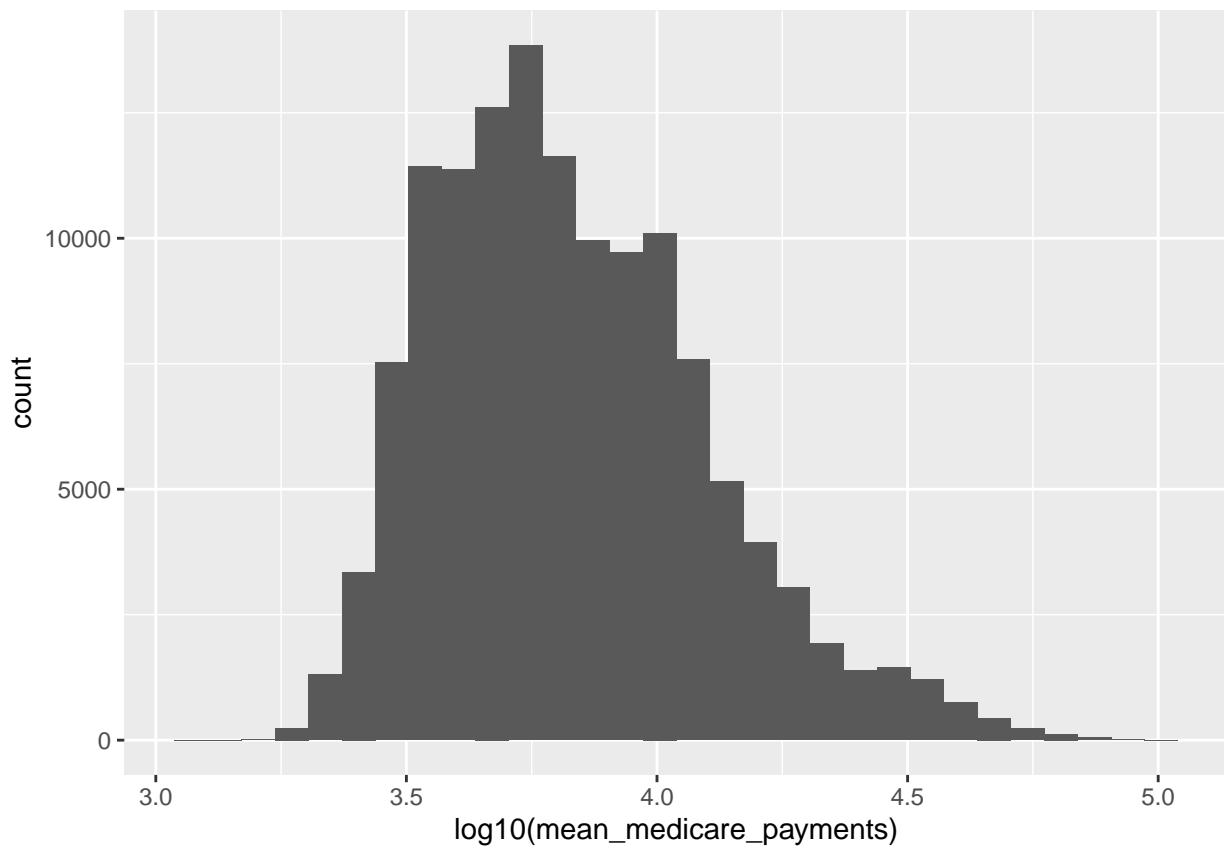


```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_medicare_payments<100000 ) %>%
  ggplot(aes(x=mean_medicare_payments))+ geom_histogram()

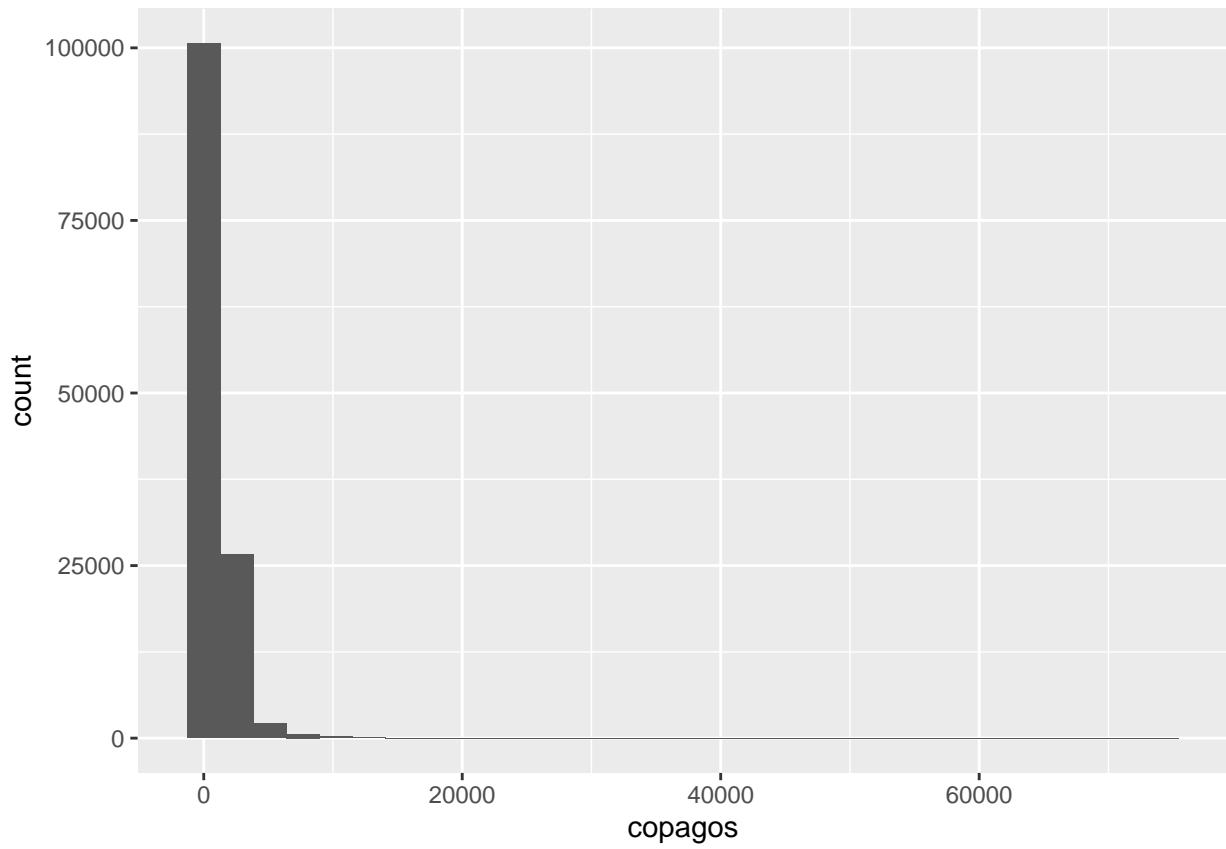
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



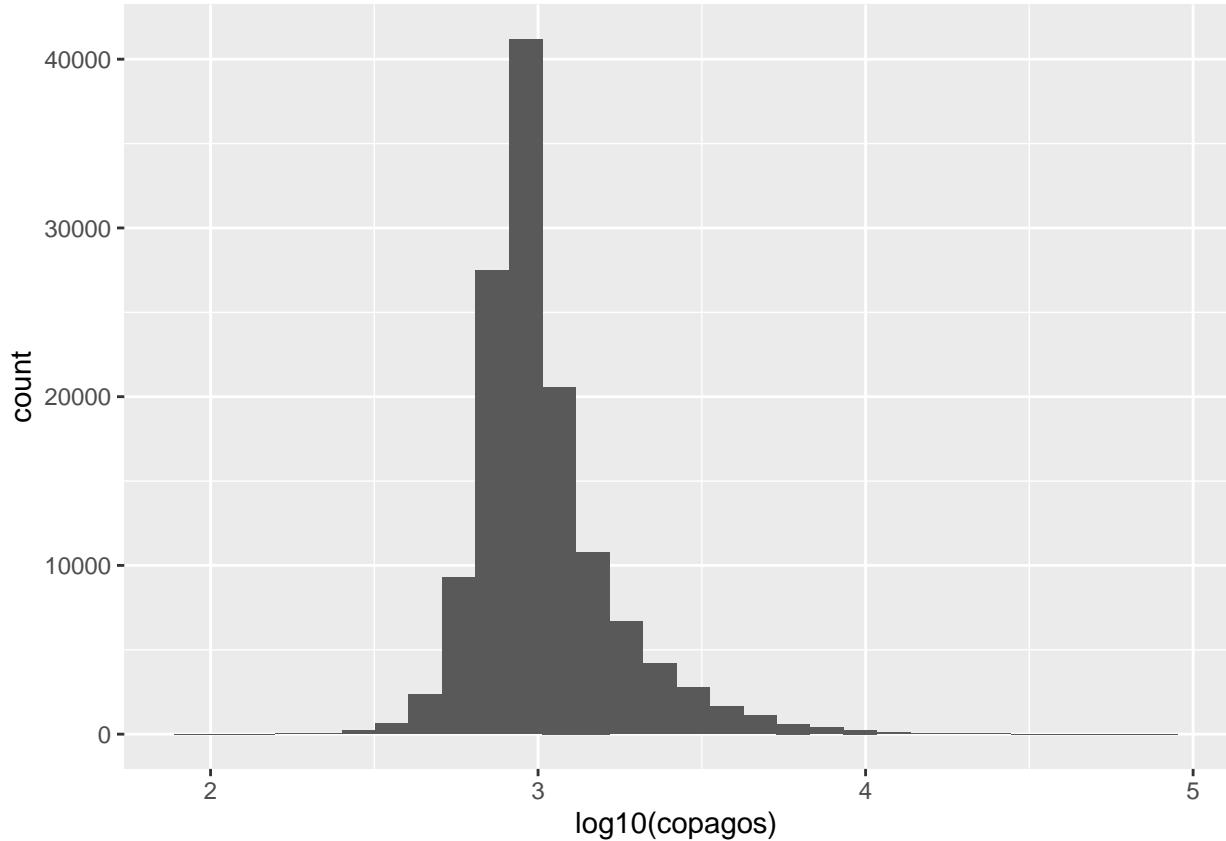
```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_medicare_payments<100000 ) %>%
  ggplot(aes(x=log10(mean_medicare_payments)))+ geom_histogram()
```



```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_medicare_payments<100000 ) %>%
  ggplot(aes(x=copagos))+ geom_histogram()
```



```
# ver si el mean total payments sigue una normal
train%>%
  filter(mean_medicare_payments<100000 ) %>%
  ggplot(aes(x=log10(copagos)))+ geom_histogram()
```



```
#train %>% select(1:14) %>%
#  na.omit() %>%
#  ggpairs(columns = 1:13, ggplot2::aes(colour=group), cardinality_threshold=50000)
```

5.4.7 Boxplot - análisis de la variables de relevancia y de los atípicos observados

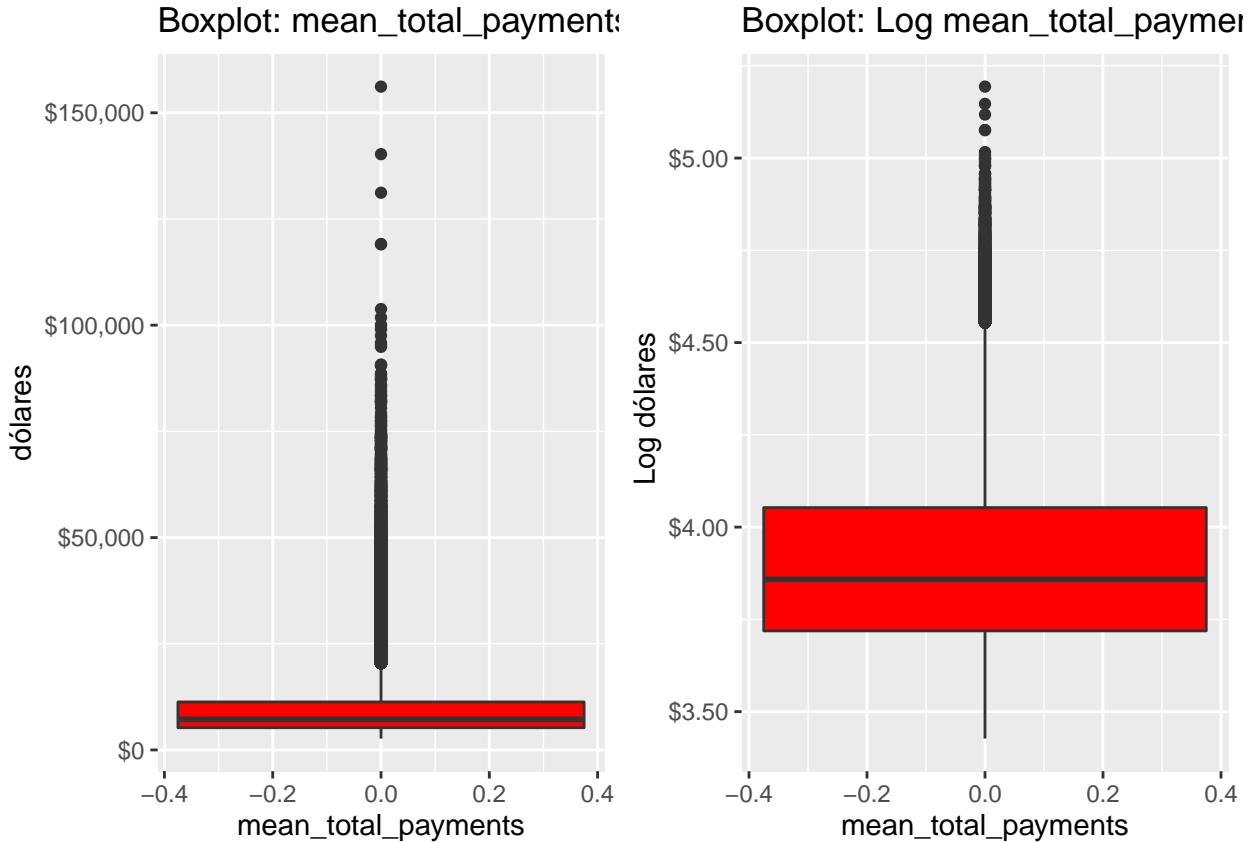
```
train_num <- train %>% select_if(is.numeric)
train_num
```

```
## # A tibble: 130,452 x 6
##   mean_total_payme~ mean_medicare_pa~ total_discharges mean_covered_ch~ copagos
##   <dbl>           <dbl>           <dbl>           <dbl>           <dbl>
## 1 11086.          8772.            NA             42249.          2314.
## 2 11853.          11076.           13             40094.          776.
## 3 47649.          44184.           17             NA             3465.
## 4 42984.          41458.           NA             187656.          1526.
## 5 5235.           4358.            52             9830.           877.
## 6 6612.           5248.            14             22260.          1364
## 7 5407.           4304.            110            NA             1103.
## 8 5160.           4043.            15             NA             1117.
## 9 8729.           6518.            70             16849.          2211.
## 10 6411.          5708.            NA             NA              703.
## # ... with 130,442 more rows, and 1 more variable: cobertura <formtbl>
```

```

p1 <- ggplot (train_num, aes(y= train_num$mean_total_payments)) + geom_boxplot(fill = "red") + scale_y_continuous(label = "dólares")
p11 <- ggplot (train_num, aes(y= log10(train_num$mean_total_payments))) + geom_boxplot(fill = "red") + scale_y_continuous(label = "Log dólares")
plot_grid(p1, p11)

```



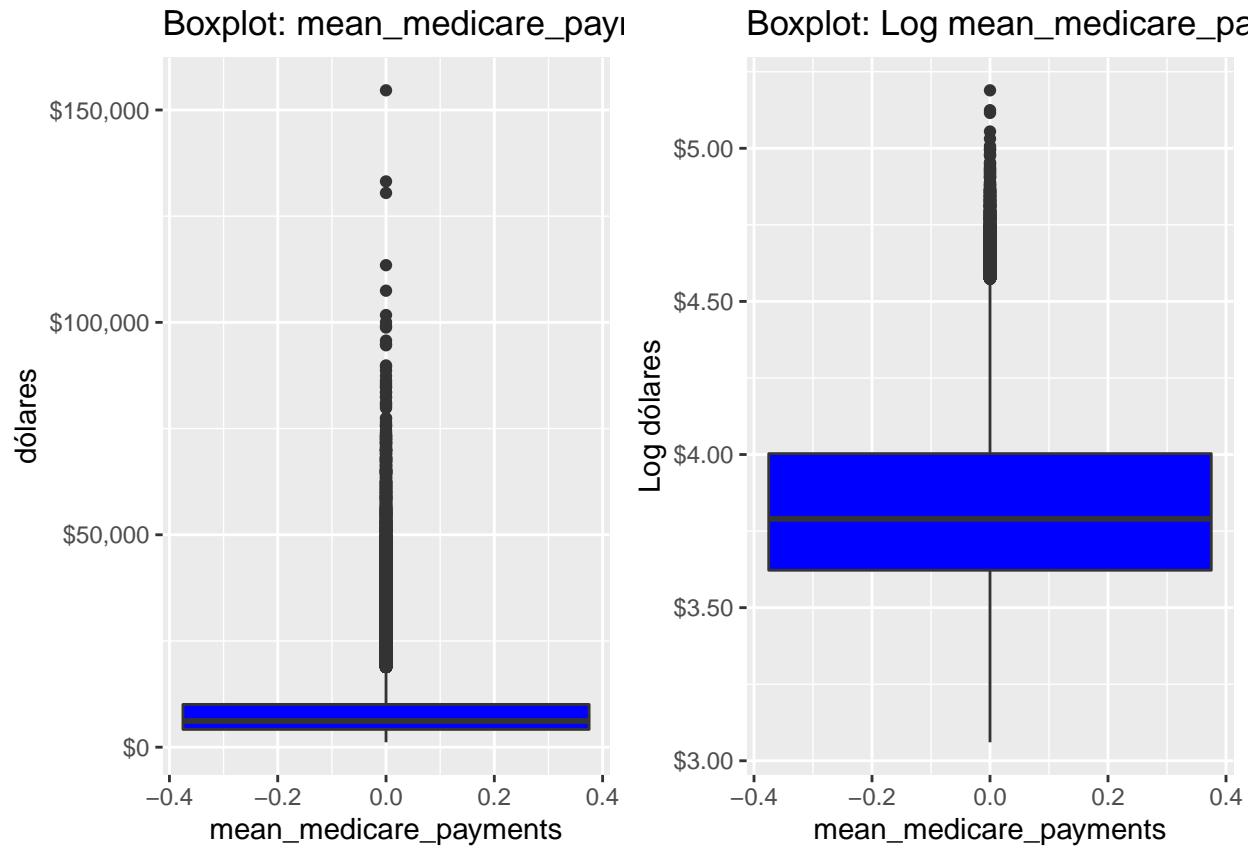
```
mean(train_num$mean_total_payments)
```

```
## [1] 9715.202
```

```

p2 <- ggplot (train_num, aes(y=train_num$mean_medicare_payments)) + geom_boxplot(fill = "blue") + scale_y_continuous(label = "dólares")
p22 <- ggplot (train_num, aes(y= log10(train_num$mean_medicare_payments))) + geom_boxplot(fill = "blue") + scale_y_continuous(label = "Log dólares")
plot_grid(p2, p22)

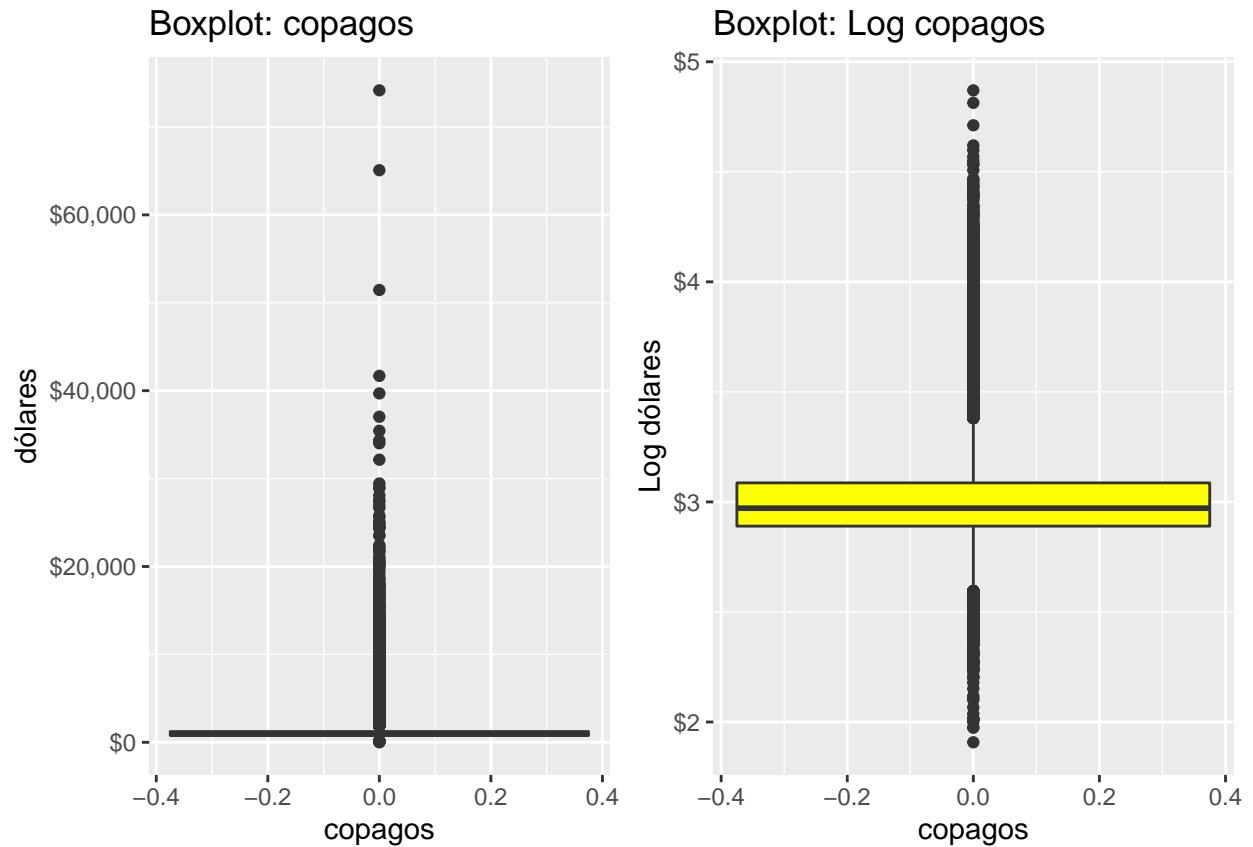
```



```
mean(train_num$mean_medicare_payments)
```

```
## [1] 8501.48
```

```
p3 <- ggplot (train_num, aes(y=train_num$copagos)) + geom_boxplot(fill = "yellow") + scale_y_continuous()
p33 <- ggplot (train_num, aes(y=log10(train_num$copagos))) + geom_boxplot(fill = "yellow") + scale_y_continuous()
plot_grid(p3, p33)
```



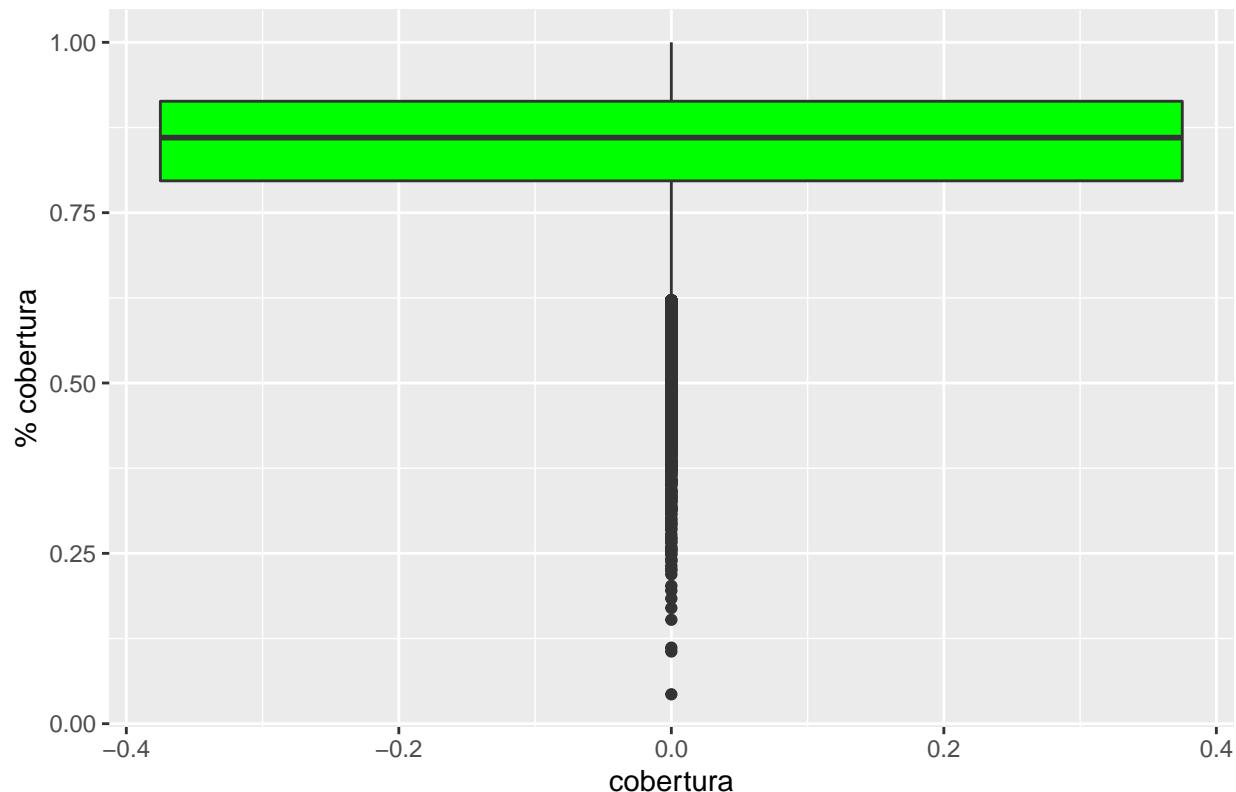
```
mean(train_num$copagos)
```

```
## [1] 1213.723
```

```
p4 <- ggplot (train_num, aes(y=train_num$cobertura)) + geom_boxplot(fill = "green") + scale_y_continuous
```

p4

Boxplot: cobertura



```
mean(train_num$cobertura)
```

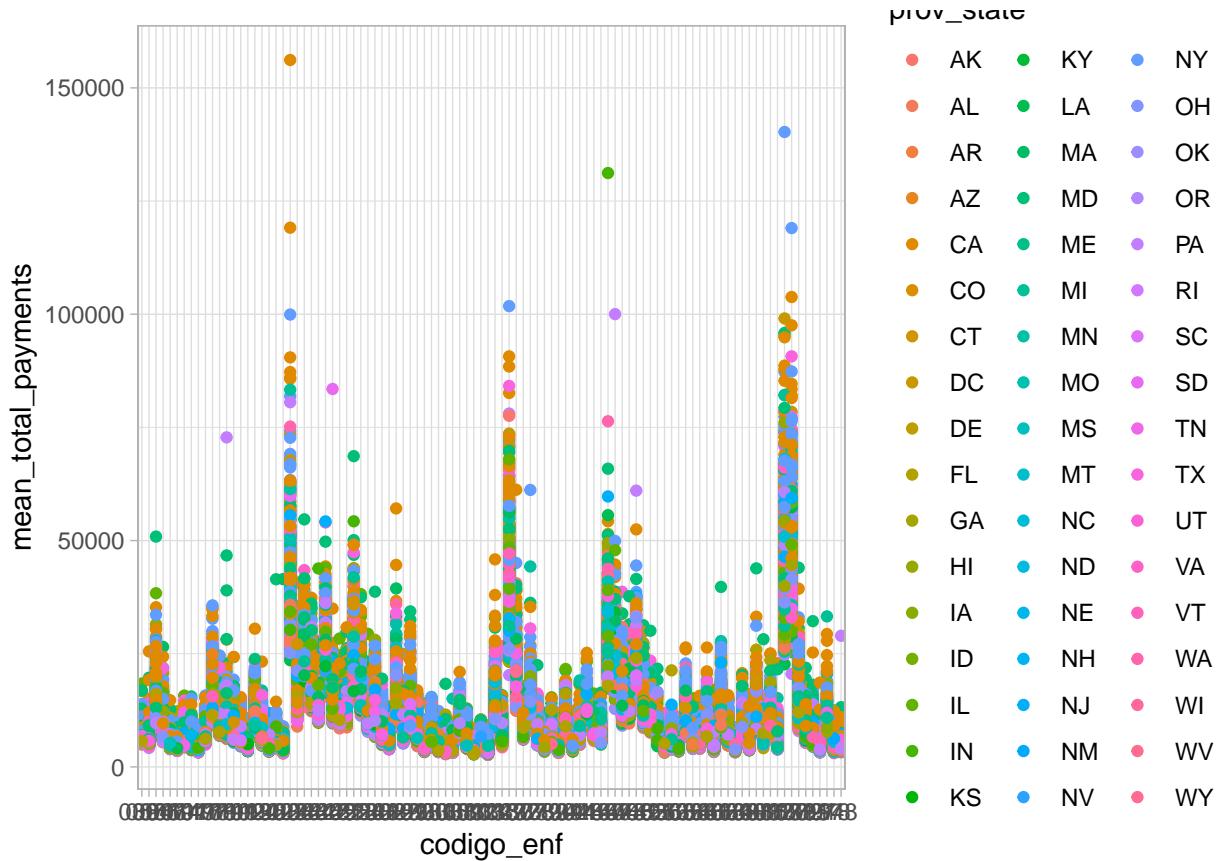
```
## [1] 84.66%
```

Regresion lineal multiple

```
library(MASS)
```

```
##  
## Attaching package: 'MASS'  
  
## The following object is masked from 'package:formattable':  
##  
##     area  
  
## The following object is masked from 'package:dplyr':  
##  
##     select  
  
#attach(train)
```

```
ggplot(train, aes(x=codigo_enf, y=mean_total_payments, color=prov_state)) +
  geom_point() + theme_light()
```



```
#https://fhernanb.github.io/libro\_regresion/varcuali.html
```

```
modelo= lm(mean_total_payments~codigo_enf+mean_medicare_payments+prov_state,data=train)
#coef(modelo)
summary(modelo)
```

```
##
## Call:
## lm(formula = mean_total_payments ~ codigo_enf + mean_medicare_payments +
##     prov_state, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -3356    -325   -127    106  71342 
##
## Coefficients:
## (Intercept) 2.041e+03  8.501e+01  24.012 < 2e-16 ***
## codigo_enf057 -3.668e+02  4.919e+01 -7.457 8.89e-14 ***
## codigo_enf064 -1.005e+02  4.647e+01 -2.162 0.030624 * 
## codigo_enf065 -2.139e+02  4.350e+01 -4.918 8.76e-07 ***
```

##	codigo_enf066	-1.650e+02	4.516e+01	-3.654	0.000258	***
##	codigo_enf069	-3.027e+02	4.448e+01	-6.805	1.01e-11	***
##	codigo_enf074	-2.280e+02	5.167e+01	-4.412	1.03e-05	***
##	codigo_enf101	-3.584e+02	4.620e+01	-7.758	8.71e-15	***
##	codigo_enf149	-2.731e+02	5.180e+01	-5.272	1.35e-07	***
##	codigo_enf176	-2.016e+02	4.744e+01	-4.249	2.14e-05	***
##	codigo_enf177	-2.514e+02	4.546e+01	-5.530	3.21e-08	***
##	codigo_enf178	-3.649e+02	4.468e+01	-8.166	3.20e-16	***
##	codigo_enf189	-3.078e+02	4.412e+01	-6.976	3.05e-12	***
##	codigo_enf190	-3.382e+02	4.241e+01	-7.974	1.54e-15	***
##	codigo_enf191	-3.806e+02	4.229e+01	-9.000	< 2e-16	***
##	codigo_enf192	-3.536e+02	4.244e+01	-8.331	< 2e-16	***
##	codigo_enf193	-3.036e+02	4.279e+01	-7.096	1.29e-12	***
##	codigo_enf194	-3.137e+02	4.167e+01	-7.529	5.16e-14	***
##	codigo_enf195	-3.139e+02	4.267e+01	-7.357	1.89e-13	***
##	codigo_enf202	-2.416e+02	4.751e+01	-5.086	3.66e-07	***
##	codigo_enf203	-1.648e+02	5.325e+01	-3.095	0.001971	**
##	codigo_enf207	1.766e+03	5.945e+01	29.708	< 2e-16	***
##	codigo_enf208	1.072e+02	4.557e+01	2.352	0.018688	*
##	codigo_enf238	1.220e+03	5.314e+01	22.963	< 2e-16	***
##	codigo_enf243	2.297e+02	5.156e+01	4.455	8.38e-06	***
##	codigo_enf244	2.801e+02	5.117e+01	5.475	4.39e-08	***
##	codigo_enf246	1.242e+03	5.558e+01	22.355	< 2e-16	***
##	codigo_enf247	1.325e+03	4.791e+01	27.662	< 2e-16	***
##	codigo_enf249	6.467e+02	5.241e+01	12.341	< 2e-16	***
##	codigo_enf251	5.735e+02	5.697e+01	10.065	< 2e-16	***
##	codigo_enf252	4.551e+02	5.238e+01	8.689	< 2e-16	***
##	codigo_enf253	3.493e+02	5.033e+01	6.941	3.91e-12	***
##	codigo_enf254	3.304e+02	5.276e+01	6.263	3.79e-10	***
##	codigo_enf280	-2.445e+02	4.526e+01	-5.402	6.60e-08	***
##	codigo_enf281	-2.938e+02	4.645e+01	-6.325	2.55e-10	***
##	codigo_enf282	-2.541e+02	5.068e+01	-5.015	5.32e-07	***
##	codigo_enf286	3.400e+02	5.354e+01	6.350	2.16e-10	***
##	codigo_enf287	1.900e+01	4.549e+01	0.418	0.676214	
##	codigo_enf291	-4.474e+02	4.253e+01	-10.519	< 2e-16	***
##	codigo_enf292	-5.496e+02	4.188e+01	-13.123	< 2e-16	***
##	codigo_enf293	-5.706e+02	4.280e+01	-13.334	< 2e-16	***
##	codigo_enf300	-3.865e+02	4.666e+01	-8.283	< 2e-16	***
##	codigo_enf301	-4.156e+02	5.345e+01	-7.776	7.55e-15	***
##	codigo_enf303	-4.905e+02	4.868e+01	-10.074	< 2e-16	***
##	codigo_enf305	-3.398e+02	5.014e+01	-6.777	1.23e-11	***
##	codigo_enf308	-3.769e+02	4.495e+01	-8.386	< 2e-16	***
##	codigo_enf309	-4.375e+02	4.319e+01	-10.130	< 2e-16	***
##	codigo_enf310	-3.540e+02	4.309e+01	-8.217	< 2e-16	***
##	codigo_enf312	-4.086e+02	4.338e+01	-9.419	< 2e-16	***
##	codigo_enf313	-4.153e+02	4.363e+01	-9.519	< 2e-16	***
##	codigo_enf314	4.963e+01	4.693e+01	1.057	0.290287	
##	codigo_enf315	-3.425e+02	5.388e+01	-6.357	2.07e-10	***
##	codigo_enf329	1.590e+03	5.696e+01	27.914	< 2e-16	***
##	codigo_enf330	6.399e+02	4.729e+01	13.530	< 2e-16	***
##	codigo_enf372	-5.152e+02	4.906e+01	-10.502	< 2e-16	***
##	codigo_enf377	-2.789e+02	4.633e+01	-6.020	1.75e-09	***
##	codigo_enf378	-3.862e+02	4.269e+01	-9.047	< 2e-16	***
##	codigo_enf379	-3.251e+02	4.738e+01	-6.861	6.88e-12	***

## codigo_enf389	-3.462e+02	4.519e+01	-7.662	1.85e-14	***
## codigo_enf390	-2.460e+02	4.739e+01	-5.192	2.08e-07	***
## codigo_enf391	-3.124e+02	4.665e+01	-6.696	2.15e-11	***
## codigo_enf392	-3.444e+02	4.183e+01	-8.232	< 2e-16	***
## codigo_enf394	-3.241e+02	4.658e+01	-6.959	3.45e-12	***
## codigo_enf418	3.537e+02	5.237e+01	6.755	1.44e-11	***
## codigo_enf419	2.579e+02	5.334e+01	4.834	1.34e-06	***
## codigo_enf439	-1.368e+02	5.205e+01	-2.628	0.008590	**
## codigo_enf460	2.756e+03	5.205e+01	52.954	< 2e-16	***
## codigo_enf469	6.345e+02	5.502e+01	11.532	< 2e-16	***
## codigo_enf470	8.692e+02	4.277e+01	20.321	< 2e-16	***
## codigo_enf473	1.400e+03	5.467e+01	25.608	< 2e-16	***
## codigo_enf480	2.911e+02	5.472e+01	5.319	1.04e-07	***
## codigo_enf481	-7.696e+01	4.450e+01	-1.729	0.083774	.
## codigo_enf482	-4.628e+01	5.127e+01	-0.903	0.366730	
## codigo_enf491	2.781e+02	5.214e+01	5.333	9.68e-08	***
## codigo_enf536	-3.966e+02	4.999e+01	-7.934	2.14e-15	***
## codigo_enf552	-2.458e+02	4.506e+01	-5.454	4.92e-08	***
## codigo_enf563	-2.748e+02	5.191e+01	-5.295	1.19e-07	***
## codigo_enf602	-2.000e+02	5.267e+01	-3.797	0.000147	***
## codigo_enf603	-2.954e+02	4.224e+01	-6.994	2.68e-12	***
## codigo_enf638	-3.984e+02	4.529e+01	-8.797	< 2e-16	***
## codigo_enf640	-4.719e+02	4.535e+01	-10.406	< 2e-16	***
## codigo_enf641	-5.020e+02	4.197e+01	-11.960	< 2e-16	***
## codigo_enf682	-2.685e+02	4.422e+01	-6.072	1.27e-09	***
## codigo_enf683	-4.421e+02	4.280e+01	-10.330	< 2e-16	***
## codigo_enf684	-4.243e+02	5.159e+01	-8.223	< 2e-16	***
## codigo_enf689	-4.704e+02	4.449e+01	-10.573	< 2e-16	***
## codigo_enf690	-4.981e+02	4.174e+01	-11.934	< 2e-16	***
## codigo_enf698	-2.060e+02	5.369e+01	-3.838	0.000124	***
## codigo_enf699	-3.852e+02	5.286e+01	-7.288	3.17e-13	***
## codigo_enf811	-3.595e+02	4.993e+01	-7.201	6.01e-13	***
## codigo_enf812	-4.644e+02	4.344e+01	-10.691	< 2e-16	***
## codigo_enf853	1.835e+03	5.904e+01	31.084	< 2e-16	***
## codigo_enf870	1.816e+03	6.548e+01	27.734	< 2e-16	***
## codigo_enf871	-1.781e+02	4.270e+01	-4.172	3.02e-05	***
## codigo_enf872	-3.160e+02	4.277e+01	-7.390	1.48e-13	***
## codigo_enf885	-3.121e+02	6.024e+01	-5.181	2.21e-07	***
## codigo_enf897	-4.763e+02	5.312e+01	-8.966	< 2e-16	***
## codigo_enf917	-1.140e+02	5.368e+01	-2.124	0.033684	*
## codigo_enf918	-4.678e+02	4.794e+01	-9.758	< 2e-16	***
## codigo_enf948	-5.151e+02	4.536e+01	-11.356	< 2e-16	***
## mean_medicare_payments	9.783e-01	1.085e-03	901.313	< 2e-16	***
## prov_stateAL	-6.103e+02	7.910e+01	-7.716	1.21e-14	***
## prov_stateAR	-6.818e+02	8.085e+01	-8.432	< 2e-16	***
## prov_stateAZ	-4.447e+02	7.957e+01	-5.588	2.30e-08	***
## prov_stateCA	-5.542e+02	7.703e+01	-7.194	6.30e-13	***
## prov_stateCO	-4.396e+02	8.109e+01	-5.422	5.92e-08	***
## prov_stateCT	-4.543e+02	8.070e+01	-5.629	1.81e-08	***
## prov_stateDC	-4.872e+02	9.369e+01	-5.200	1.99e-07	***
## prov_stateDE	-3.487e+02	9.730e+01	-3.584	0.000338	***
## prov_stateFL	-6.168e+02	7.736e+01	-7.972	1.57e-15	***
## prov_stateGA	-5.077e+02	7.833e+01	-6.482	9.10e-11	***
## prov_stateHI	1.691e+02	9.457e+01	1.788	0.073746	.

```

## prov_stateIA      -5.996e+02 8.156e+01 -7.351 1.97e-13 ***
## prov_stateID     -4.478e+02 9.127e+01 -4.906 9.32e-07 ***
## prov_stateIL     -3.449e+02 7.762e+01 -4.444 8.84e-06 ***
## prov_stateIN     -4.973e+02 7.864e+01 -6.324 2.56e-10 ***
## prov_stateKS     -5.556e+02 8.192e+01 -6.782 1.19e-11 ***
## prov_stateKY     -6.465e+02 7.927e+01 -8.156 3.50e-16 ***
## prov_stateLA     -5.376e+02 7.948e+01 -6.764 1.34e-11 ***
## prov_stateMA     -6.596e+02 7.871e+01 -8.380 < 2e-16 ***
## prov_stateMD     -5.507e+02 7.902e+01 -6.969 3.21e-12 ***
## prov_stateME     -6.150e+02 8.599e+01 -7.152 8.57e-13 ***
## prov_stateMI     -6.453e+02 7.811e+01 -8.262 < 2e-16 ***
## prov_stateMN     -4.220e+02 8.043e+01 -5.247 1.55e-07 ***
## prov_stateMO     -6.323e+02 7.872e+01 -8.032 9.65e-16 ***
## prov_stateMS     -6.444e+02 8.014e+01 -8.041 8.99e-16 ***
## prov_stateMT     -5.441e+02 9.301e+01 -5.850 4.92e-09 ***
## prov_stateNC     -6.481e+02 7.827e+01 -8.280 < 2e-16 ***
## prov_stateND     -7.397e+02 9.567e+01 -7.732 1.07e-14 ***
## prov_stateNE     -4.372e+02 8.473e+01 -5.159 2.48e-07 ***
## prov_stateNH     -5.811e+02 8.662e+01 -6.709 1.97e-11 ***
## prov_stateNJ     -6.487e+02 7.825e+01 -8.290 < 2e-16 ***
## prov_stateNM     -4.086e+02 8.574e+01 -4.765 1.89e-06 ***
## prov_stateNV     -2.410e+02 8.343e+01 -2.889 0.003868 **
## prov_stateNY     -4.977e+02 7.734e+01 -6.436 1.23e-10 ***
## prov_stateOH     -6.220e+02 7.782e+01 -7.992 1.34e-15 ***
## prov_stateOK     -6.056e+02 8.001e+01 -7.569 3.79e-14 ***
## prov_stateOR     -3.650e+02 8.307e+01 -4.393 1.12e-05 ***
## prov_statePA     -5.571e+02 7.765e+01 -7.175 7.29e-13 ***
## prov_stateRI     -5.261e+02 9.057e+01 -5.809 6.32e-09 ***
## prov_stateSC     -4.917e+02 7.965e+01 -6.173 6.70e-10 ***
## prov_stateSD     -3.387e+02 9.304e+01 -3.640 0.000273 ***
## prov_stateTN     -5.547e+02 7.871e+01 -7.047 1.84e-12 ***
## prov_stateTX     -5.007e+02 7.728e+01 -6.479 9.26e-11 ***
## prov_stateUT     1.038e+02 8.782e+01 1.182 0.237180
## prov_stateVA     -4.191e+02 7.858e+01 -5.333 9.66e-08 ***
## prov_stateVT     -4.040e+02 1.059e+02 -3.814 0.000137 ***
## prov_stateWA     -3.061e+02 7.966e+01 -3.843 0.000122 ***
## prov_stateWI     -4.912e+02 7.965e+01 -6.168 6.96e-10 ***
## prov_stateWV     -6.824e+02 8.198e+01 -8.324 < 2e-16 ***
## prov_stateWY     2.251e+02 1.079e+02 2.086 0.037021 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1049 on 130301 degrees of freedom
## Multiple R-squared:  0.9814, Adjusted R-squared:  0.9814
## F-statistic: 4.581e+04 on 150 and 130301 DF,  p-value: < 2.2e-16

anova(modelo)

## Analysis of Variance Table
##
## Response: mean_total_payments
##                               Df   Sum Sq   Mean Sq   F value   Pr(>F)
## codigo_enf                  99 6.4490e+12 6.5141e+10 5.9233e+04 < 2.2e-16 ***
## mean_medicare_payments       1 1.1062e+12 1.1062e+12 1.0059e+06 < 2.2e-16 ***

```

```
## prov_state      50 1.7506e+09 3.5011e+07 3.1836e+01 < 2.2e-16 ***
## Residuals       130301 1.4330e+11 1.0998e+06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```