TP M3 M4

Tomás Emilio Baigorria e Iván Robles Urquiza

2024-02-12

# Preparación y exploración de datos

### Importación de librerías.

library(tidyverse)  
library(tidymodels)  
tidymodels\_prefer()  
library(GGally)  
library(gtsummary)  
library(gt)  
library(lubridate)  
library(ggplot2)  
library(scales)  
library(paletteer)  
library(viridis)  
library(corrplot)  
library(hrbrthemes)  
library(foreach)  
library(rpart.plot)  
library(car)  
library(baguette)

### Preparación del dataset.

* Renombramos las variables codificadas y eliminamos las columnas que consideramos irrelevantes (CODUSU, y Nro. de Hogar).

data\_eph <- read\_csv("./Data/M34\_202103\_eph.csv") %>%   
 rename("ROL\_HOGAR" = "CH03") %>%   
 rename("SEXO" = "CH04") %>%   
 rename("EDAD" = "CH06") %>%  
 rename("ESTADO CONYUGAL" = "CH07") %>%  
 rename("SECTOR\_PUB\_PRIV" = "PP04A") %>%  
 rename("CANT\_HORAS" = "PP3E\_TOT") %>%  
 rename("INGRESO\_OCUP\_PRINCIPAL" = "P21") %>%   
 rename("ESTADO\_CONYUGAL" = "ESTADO CONYUGAL") %>%   
 select(3:15) %>%   
 relocate(INGRESO\_OCUP\_PRINCIPAL)

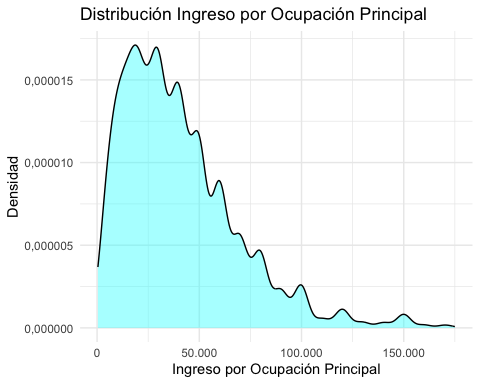
###Exploración de los datos

* Comenzamos con el estudio de la distribución de nuestra variable objetivo, el ingreso percibido por la ocupación principal:
  + En una primera iteración del gráfico observamos la presencia de un outlier, el cual filtramos excluyendo de la muestra el percentil superior.
  + Observamos también la existencia de valores negativos (-9) y nulos, que corresponden a “no respuesta” y a “no les corresponde la secuencia alcanzada”, respectivamente. Decidimos quitarlos para obtener datos más homogéneos.
  + Luego elaboramos un gráfico de densidad.

top10ingresos <- top\_n(data\_eph %>% select(INGRESO\_OCUP\_PRINCIPAL), 10) %>%   
 arrange(desc(INGRESO\_OCUP\_PRINCIPAL))

## Selecting by INGRESO\_OCUP\_PRINCIPAL

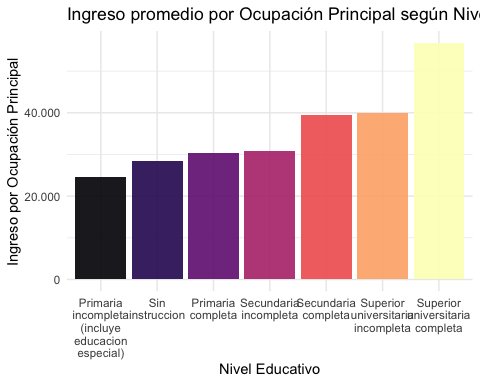
data\_eph\_limpio <- data\_eph %>% filter(INGRESO\_OCUP\_PRINCIPAL > 0) %>%   
 filter(INGRESO\_OCUP\_PRINCIPAL < quantile(INGRESO\_OCUP\_PRINCIPAL, .99))



- Como curiosidad, notamos la existencia de picos de densidad en los números "redondos", los cuales asociamos tanto a un sesgo en los encuestados al redondear su salario como a un posible sesgo en los empleadores a la hora de definir los salarios de sus empleados.

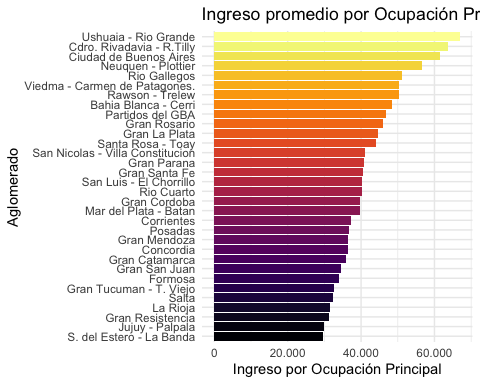
## Análisis bivariado y multivariado

A continuación, estudiamos la relación entre el ingreso (considerada variable principal a explicar) y algunas de las variables categóricas presentes en el dataset.



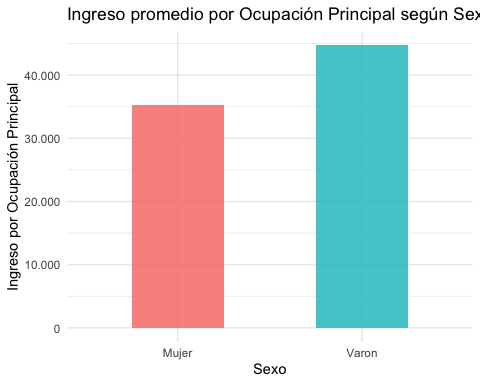
- Como vemos, existe una relación clara entre el nivel educativo de las personas y su remuneración percibida.  
  
- Lo mismo ocurre con algunas de las demás variables categóricas: Aglomerado, Sexo, Categoría Ocupacional y Categoría/sector.

### Ingreso por aglomerado, como variable geográfica

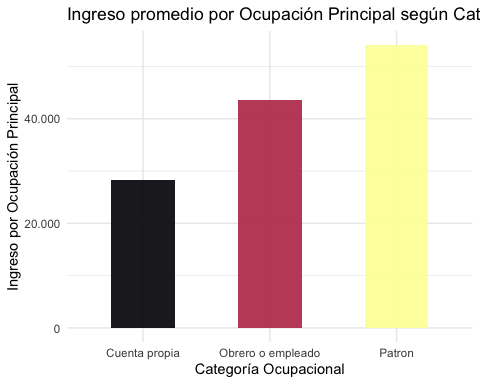


La distribución media de los ingresos por aglomerado nos permite ver aquellas diferencias que surjen del agrupamiento geográfico por las zonas del relevamiento. Se presentan grandes diferencias entre los valores mínimos y máximos.

### Distinción por sexo

 - Se presenta una considerable brecha de ingresos por ocupación principal cuando distinguimos por Sexo, lo que nos advierte que se trata de una variable relevante para la predicción.

### Ingreso medio por categoría ocupacional

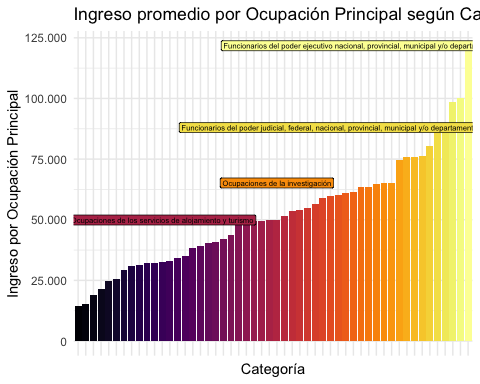
 - La visualización demuestra que la categoría ocupacional es otra variable a ponderar en la predicción del ingreso a partir de la gran diferencia que se halla entre sus medias.

Procedemos a profundizar en el análisis de las distintas subcategorías para percibir si existe variabilidad dentro de cada categoría ocupacional.

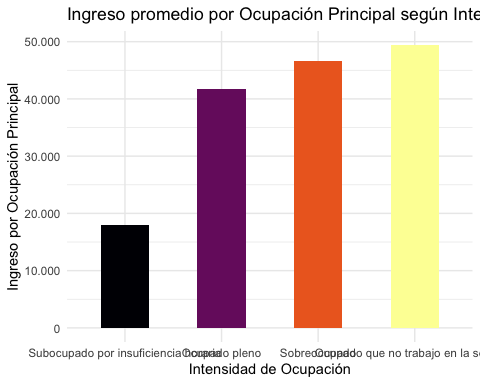
## # A tibble: 10 × 2  
## CATEGORIA ingreso\_promedio  
## <fct> <dbl>  
## 1 Ocupaciones de los servicios domésticos 14285.  
## 2 Ocupaciones de la comercialización ambulante y callejera 15236.  
## 3 Ocupaciones del cuidado y la atención de personas 19069.  
## 4 Ocupaciones de los servicios sociales varios 21328.  
## 5 Ocupaciones de la producción forestal 24673.  
## 6 Ocupaciones de servicios gastronómicos 25671.  
## 7 Ocupaciones del deporte 29365   
## 8 Ocupaciones de la producción agrícola 30847.  
## 9 Ocupaciones de servicios de limpieza no domésticos 31230.  
## 10 Ocupaciones de la reparación de bienes de consumo 31984.

## # A tibble: 10 × 2  
## CATEGORIA ingreso\_promedio  
## <fct> <dbl>  
## 1 Ocupaciones de la asesoria y la consultaría 74460   
## 2 Ocupaciones de la producción de software 75710   
## 3 Directivos de medianas empresas privadas productoras de bie… 75883.  
## 4 Ocupaciones de la gestión jurídico-legal 76151.  
## 5 Directivos de organismos, empresas e instituciones estatales 80485.  
## 6 Funcionarios del poder judicial, federal, nacional, provinc… 88000   
## 7 Directivos de grandes empresas privadas productoras de bien… 89806.  
## 8 Directivos de instituciones sociales 98333.  
## 9 Ocupaciones de la producción extractiva 100094.  
## 10 Funcionarios del poder ejecutivo nacional, provincial, muni… 121750

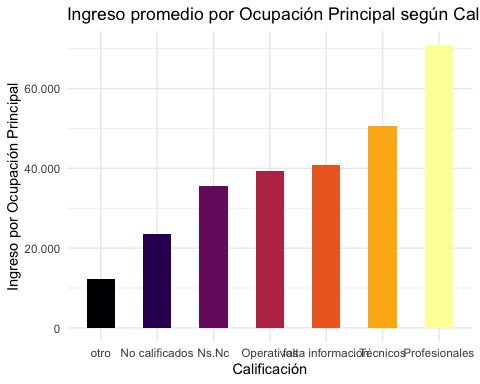
## # A tibble: 3 × 2  
## CATEGORIA ingreso\_promedio  
## <fct> <dbl>  
## 1 Ocupaciones de los servicios de alojamiento y turismo 49889.  
## 2 Ocupaciones de la investigación 65157.  
## 3 Funcionarios del poder judicial, federal, nacional, provinci… 88000



### Intensidad de ocupación



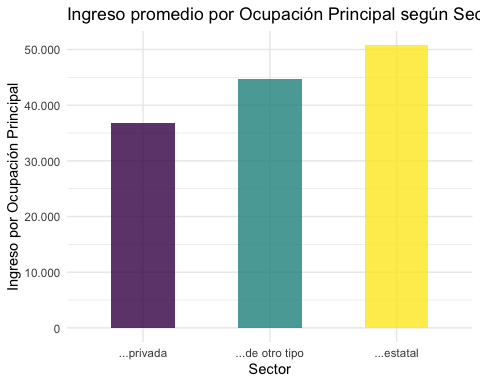
- En el caso de Intensidad de Ocupación, vemos por un lado la existencia de una variable correspondiente a los Ocupados que declaran no haber trabajado durante la semana previa a la encuesta. Más adelante removeremos estos casos para mejorar el modelo.  
  
- Por otro lado, notamos que existe solo una magra diferencia en los promedios salariales de quienes declaran ser ocupados plenos y quienes declaran estar sobreocupados.  
  
- Por último, creemos que es posible capturar de mejor manera el factor horas de trabajo mediante la variable numérica correspondiente, la cual analizaremos en detalle posteriormente.



datos\_imprecisos <- data\_eph\_limpio %>% select(CALIFICACION) %>%   
 filter(CALIFICACION %in% c("otro", "Ns.Nc", "falta informacion"))  
  
count(datos\_imprecisos)

## # A tibble: 1 × 1  
## n  
## <int>  
## 1 84

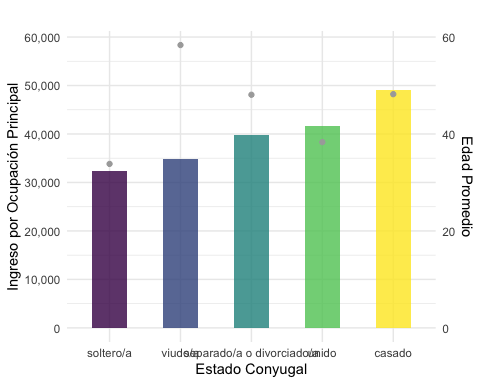
- La variable de calificación cuenta con algunos sesgos que nos impiden ponderar de forma eficiente sus resultados como función del ingreso. Sin embargo, notamos que aquellos que han sido identificados "Profesionales" cuentan con mayores ingresos que el resto (entre los que se encuentran los Ns. Nc., otros, y los faltos de información).  
  
- Si bien se trata de sólo 84 valores, consideramos que la cuestión de la calificación y la formación de los encuestados puede ser capturada más nítidamente mediante el nivel educativo.



- Para el caso de la ocupación principal, consideramos:  
1. la categoría "de otro tipo" no es lo suficientemente clara como para ser relevante y   
2. la variable de sector/categoría captura el mismo elemento con mayor claridad.

### Estado conyugal

## [1] "unido?" "soltero/a?"   
## [3] "separado/a o divorsiado/a?" "casado?"   
## [5] "viudo/a?"



- Finalmente, tenemos el Estado Conyugal del individuo. Creemos que no es relevante para el análisis dada la correlación existente entre la edad y el estado conyugal de las personas, la cual visualizamos en el gráfico.  
  
- Analicemos ahora la relación de nuestra variable de estudio con las variables numéricas del data set: Cantidad de Horas de Trabajo y Edad.

max(data\_eph\_limpio$CANT\_HORAS)

## [1] 999

min(data\_eph\_limpio$CANT\_HORAS)

## [1] 0

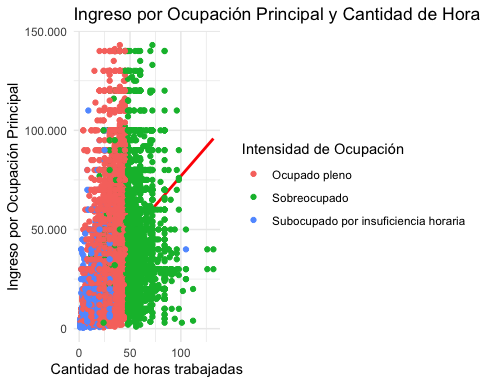
count(data\_eph\_limpio %>% filter(CANT\_HORAS == 0))

## # A tibble: 1 × 1  
## n  
## <int>  
## 1 654

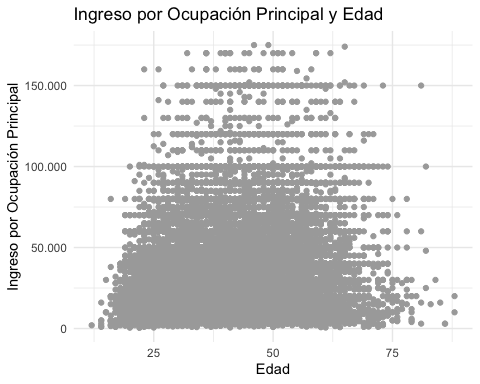
count(data\_eph\_limpio %>% filter(INTENSI == "Ocupado que no trabajo en la semana"))

## # A tibble: 1 × 1  
## n  
## <int>  
## 1 651

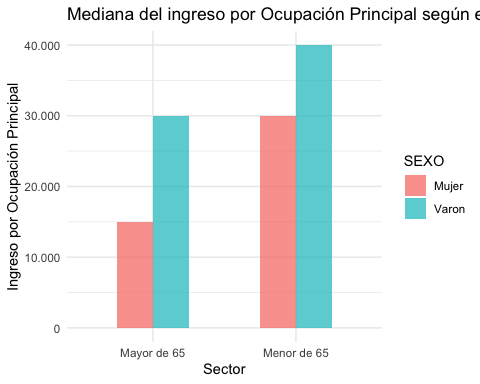
- En primer lugar, observamos la existencia de valores absurdos (999 horas trabajadas) en nuestra primer variable, los cuales limpiamos previo a graficar.  
  
- Asimismo, vemos una variedad de registros con ingresos mayores a cero y horas trabajadas nulas que corresponden a individuos que declaran no haber trabajado la semana previa (generalmente debido a vacaciones, enfermedad, etc.). Decidimos filtrar estos casos para mejorar la performance del modelo.



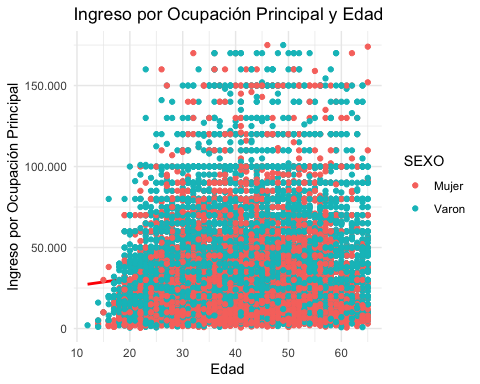
- El gráfico da cuenta de una relación clara entre las dos variables. En el caso de la edad, la relación es un tanto más difusa:

 -> Concluímos que, por cómo se distribuyen los datos para la variable *Intensidad de Ocupación*, esta podría ser removida de nuestra regresión en tanto la variable *Cantidad de Horas Trabajadas* cumple un rol similar.

En este caso, vemos que la distribución de los datos se asemeja a una campana de gauss. A priori, consideramos que ello puede estar vinculado con la ocurrencia de menores ingresos pasada la edad jubilatoria. Así lo visualizamos en el siguiente gráfico:



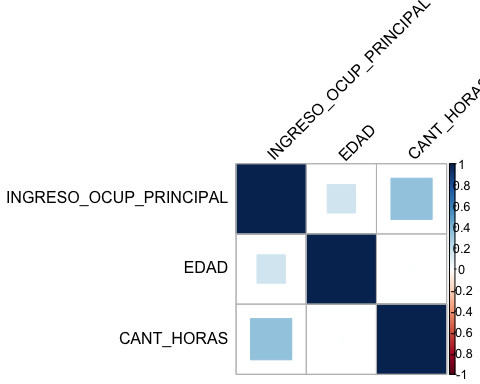
- Usamos la mediana para atenuar el peso de los outliers y de la disparidad en el número de registros presnetes en cada subgrupo.   
  
- En base a lo visto, creemos que puede ser una buena idea controlar la variable edad por la edad jubilatoria y el sexo.



## Correlación

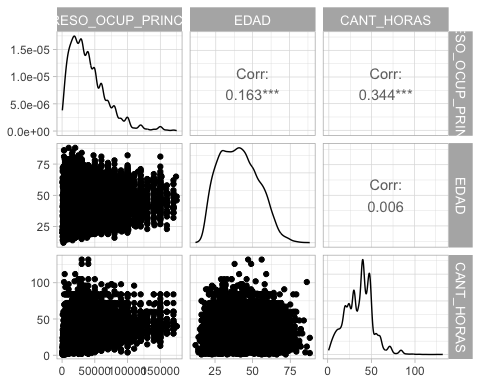
Proponemos una matriz de correlación con aquellas variables de tipo numéricas.

data\_eph\_limpio %>%  
 select\_if(is.numeric) %>%  
 cor() %>%  
 corrplot(method = "square", tl.col = "black", tl.srt = 45)



Proponemos como primer aproximaxión la siguiente matriz descriptiva para visualizar distribuciones entre las variables de tipo numéricas.

data\_eph\_limpio %>%  
 select\_if(is.numeric) %>%  
 ggpairs(labeller = label\_wrap\_gen(width = 5)) +  
 theme\_light() +  
 theme(strip.text.x = element\_text(size = 10),  
 strip.text.y = element\_text(size = 10))



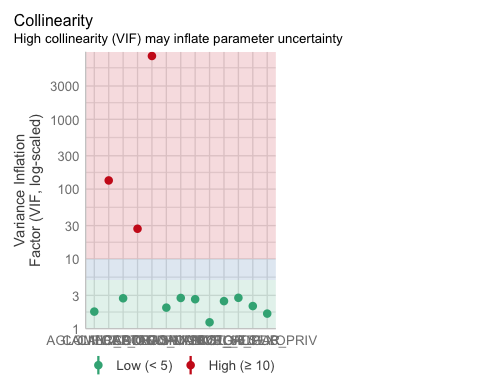
## Modelización

Para corroborar la multicolinealidad que podría afectar a nuestro modelo de regresión, planteamos un modelo simple para nuestra variable explicada INGRESO\_OCUP\_PRINCIPAL, utlizamos la función vif():

modelo\_lineal <- lm(INGRESO\_OCUP\_PRINCIPAL ~ ., data = data\_eph\_limpio)  
  
vif(modelo\_lineal)

## GVIF Df GVIF^(1/(2\*Df))  
## AGLOMERADO 1.757112 31 1.009133  
## NIVEL\_ED 2.481970 6 1.078698  
## ROL\_HOGAR 2.769476 9 1.058224  
## SEXO 1.644535 1 1.282394  
## EDAD 2.002700 1 1.415168  
## ESTADO\_CONYUGAL 2.754788 4 1.135040  
## SECTOR\_PUB\_PRIV 2.109367 2 1.205141  
## CAT\_OCUP 27.035504 2 2.280256  
## INTENSI 2.643341 2 1.275082  
## CANT\_HORAS 2.725860 1 1.651018  
## CATEGORIA 8073.552158 51 1.092206  
## CALIFICACION 133.043070 6 1.503140  
## MAYOR\_65 1.234197 1 1.110944

p <- performance::check\_model(modelo\_lineal, check = 'vif') %>%  
 plot() +  
 theme(axis.text.x = element\_blank())

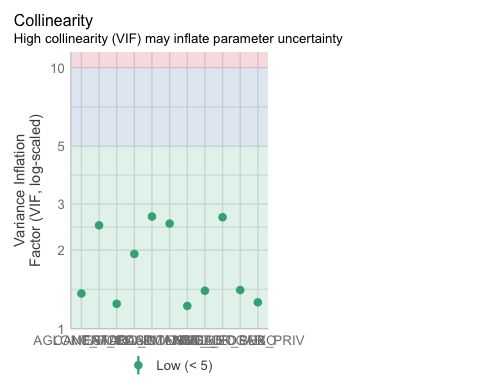


A partir de estos datos, podemos descartar del modelo aquellas variables que presentan multicolinealidad. Vemos que el problema se presenta para algunas de nuestras variables, por lo que procedemos a remover del dataset aquellas dos que más perjudicarían al modelo. Luego, analizamos multicolinealidad nuevamente.

data\_eph\_lm = subset(data\_eph\_limpio, select = -c(CALIFICACION, CATEGORIA))  
  
modelo\_lineal <- lm(INGRESO\_OCUP\_PRINCIPAL ~ ., data = data\_eph\_lm)  
  
vif(modelo\_lineal)

## GVIF Df GVIF^(1/(2\*Df))  
## AGLOMERADO 1.363882 31 1.005018  
## NIVEL\_ED 1.397332 6 1.028273  
## ROL\_HOGAR 2.671846 9 1.056116  
## SEXO 1.261647 1 1.123231  
## EDAD 1.933323 1 1.390440  
## ESTADO\_CONYUGAL 2.689651 4 1.131650  
## SECTOR\_PUB\_PRIV 1.406365 2 1.088991  
## CAT\_OCUP 1.245513 2 1.056421  
## INTENSI 2.527342 2 1.260858  
## CANT\_HORAS 2.488102 1 1.577372  
## MAYOR\_65 1.222030 1 1.105455

p <- performance::check\_model(modelo\_lineal, check = 'vif') %>%  
 plot() +  
 theme(axis.text = element\_text(angle = 90))



- Anulado el problema de multicolinealidad para las variables que formarán parte de nuestro modelo, procedemos a analizar significatividad para las variables restantes.

En primer lugar, aplicamos la conversión a dummy de variables categóricas como recipe para preprocesamiento.Luego, hacemos un summary() donde prestamos especial importancia al test de significatividad.

Este método nos resultará útil a la hora de correr nuestro modelo, aunque por la cantidad de valores que puede tomar la variable aglomerado complica la visualización en formato tabla tras la conversión.

# Conversión a dummy de variables categóricas como recipe para preprocesamiento.  
  
recipe\_lm <- recipe(INGRESO\_OCUP\_PRINCIPAL ~ ., data = data\_eph\_lm) %>%  
 step\_dummy(all\_nominal(), -all\_outcomes())  
  
  
recipe\_lm %>%   
 prep() %>%   
 juice() %>%   
 head()

## # A tibble: 6 × 61  
## EDAD CANT\_HORAS INGRESO\_OCUP\_PRINCIPAL AGLOMERADO\_Cdro..Rivadavia...R.Tilly  
## <dbl> <dbl> <dbl> <dbl>  
## 1 32 48 20000 0  
## 2 30 30 8000 0  
## 3 47 48 11000 0  
## 4 25 36 19000 0  
## 5 59 84 70000 0  
## 6 30 61 45000 0  
## # ℹ 57 more variables: AGLOMERADO\_Ciudad.de.Buenos.Aires <dbl>,  
## # AGLOMERADO\_Concordia <dbl>, AGLOMERADO\_Corrientes <dbl>,  
## # AGLOMERADO\_Formosa <dbl>, AGLOMERADO\_Gran.Catamarca <dbl>,  
## # AGLOMERADO\_Gran.Cordoba <dbl>, AGLOMERADO\_Gran.La.Plata <dbl>,  
## # AGLOMERADO\_Gran.Mendoza <dbl>, AGLOMERADO\_Gran.Parana <dbl>,  
## # AGLOMERADO\_Gran.Resistencia <dbl>, AGLOMERADO\_Gran.Rosario <dbl>,  
## # AGLOMERADO\_Gran.San.Juan <dbl>, AGLOMERADO\_Gran.Santa.Fe <dbl>, …

lmodel <- linear\_reg() %>%  
 set\_engine("lm")  
  
lm\_workflow <- workflow() %>%  
 add\_model(lmodel) %>%  
 add\_recipe(recipe\_lm)  
  
lm\_fit <- lm\_workflow %>%  
 fit(data = data\_eph\_lm)  
  
tidy(lm\_fit) %>%   
 select(term,p.value) %>%   
 filter(p.value > 0.05) %>%  
 arrange(p.value)

## # A tibble: 11 × 2  
## term p.value  
## <chr> <dbl>  
## 1 NIVEL\_ED\_Primaria.incompleta..incluye.educacion.especial. 0.0538  
## 2 SECTOR\_PUB\_PRIV\_...estatal 0.0702  
## 3 ROL\_HOGAR\_Madre.Padre 0.0859  
## 4 ROL\_HOGAR\_Otros.Familiares 0.145   
## 5 NIVEL\_ED\_Sin.instruccion 0.196   
## 6 ROL\_HOGAR\_Suegro.a 0.231   
## 7 AGLOMERADO\_Mar.del.Plata...Batan 0.291   
## 8 AGLOMERADO\_Santa.Rosa...Toay 0.391   
## 9 AGLOMERADO\_Viedma...Carmen.de.Patagones. 0.452   
## 10 AGLOMERADO\_Gran.Rosario 0.929   
## 11 AGLOMERADO\_Partidos.del.GBA 0.938

Notamos que que algunos de los valores que adopta *AGLOMERADOS* tras su conversión a dummy no son relevantes para el modelo en tanto el p-valor para el test de significatividad individual es suficientemente alto para rechazar. Por esto, decidimos retirar la variable explicativa de nuestro modelo ajustado.

Por otro lado, la variable ROL\_HOGAR demuestra lo mismo para algunos de sus valores. Y como se ha visto anteriormente, la brecha de ingresos medios que presenta el valor Jefe/a de hogar con respecto al resto es suficientemente alto como para distinguir este valor del resto. Para nuestro modelo, simplificaremos la dummy a sólo 2 valores, de forma que la variable se ajuste a su forma más significativa.

Por último, notamos que los valores correspondientes a *NIVEL\_ED* Sin intrucción son descartables por el nulo significado para el contexto de la investigación y por la practicamente nula participación relativa en el total de la muestra.

data\_eph\_lm\_ajusted <- data\_eph\_lm %>%  
 filter(NIVEL\_ED != "Sin instruccion") %>%  
 select(-AGLOMERADO, -SECTOR\_PUB\_PRIV) %>%   
 mutate(ROL\_HOGAR = ifelse(ROL\_HOGAR != "Jefe/a", "Otro", ROL\_HOGAR))  
  
head(data\_eph\_lm\_ajusted)

## # A tibble: 6 × 10  
## INGRESO\_OCUP\_PRINCIPAL NIVEL\_ED ROL\_HOGAR SEXO EDAD ESTADO\_CONYUGAL CAT\_OCUP  
## <dbl> <chr> <chr> <chr> <dbl> <chr> <chr>   
## 1 20000 Secunda… Otro Varon 32 unido Obrero …  
## 2 8000 Secunda… Jefe/a Mujer 30 soltero/a Cuenta …  
## 3 11000 Secunda… Otro Mujer 47 separado/a o d… Patron   
## 4 19000 Secunda… Otro Mujer 25 soltero/a Obrero …  
## 5 70000 Secunda… Otro Varon 59 casado Obrero …  
## 6 45000 Superio… Jefe/a Varon 30 soltero/a Obrero …  
## # ℹ 3 more variables: INTENSI <chr>, CANT\_HORAS <dbl>, MAYOR\_65 <chr>

Con la nueva base ajustada, pasamos a modelar nuevamente.

recipe\_lm\_ajusted <- recipe(INGRESO\_OCUP\_PRINCIPAL ~ . , data = data\_eph\_lm\_ajusted) %>%  
 step\_interact(terms = ~ SEXO:CANT\_HORAS + INTENSI:CANT\_HORAS + EDAD:CANT\_HORAS + EDAD:INTENSI) %>%  
 step\_dummy(all\_nominal(), -all\_outcomes())  
  
lm\_workflow <- workflow() %>%  
 add\_model(lmodel) %>%  
 add\_recipe(recipe\_lm\_ajusted)  
  
lm\_fit <- lm\_workflow %>%  
 fit(data = data\_eph\_lm\_ajusted)  
  
tidy(lm\_fit) %>%   
 select(term,p.value) %>%   
 filter(p.value > 0.13) %>%  
 arrange(p.value)

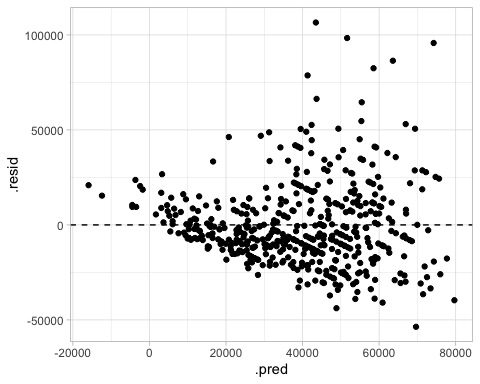
## # A tibble: 1 × 2  
## term p.value  
## <chr> <dbl>  
## 1 SEXOVaron\_x\_CANT\_HORAS 0.268

Sin contar el valor puntual de la interacción que califica la cantidad de horas y sexo varón (que por motivos que hemos explayado en el análisis exploratorio consideramos relevante), no tenemos otros resultados para variables no rechazadas en el test de significatividad. Concluimos que podemos proceder al análisis del modelo que hemos planteado.

### Análisis del modelo y predicciones

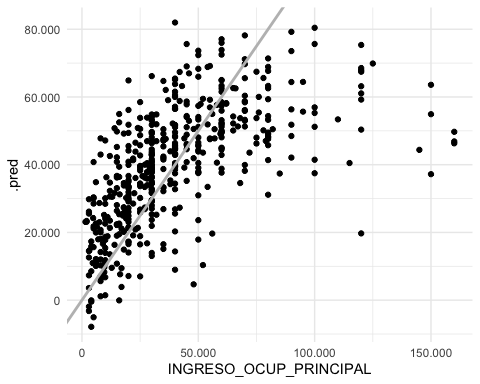
A partir de una muestra de los errores, plantearemos una visualización que nos permita detectar cómo se distribuyen los mismos.

sample\_size <- 500 # Por ejemplo, 500 puntos  
  
# Tomar una muestra aleatoria de tamaño\_muestra de los residuos  
residuos\_muestra <- augment(lm\_fit, new\_data = data\_eph\_lm\_ajusted) %>%  
 mutate(.resid = INGRESO\_OCUP\_PRINCIPAL - .pred) %>%  
 select(.resid, .pred) %>%  
 sample\_n(size = sample\_size, replace = FALSE) # sample\_n() para tomar la muestra aleatoria  
  
# Graficar los residuos de la muestra aleatoria  
ggplot(residuos\_muestra, aes(x = .pred, y = .resid)) +  
 geom\_point() +  
 geom\_hline(yintercept = 0, linetype = 'dashed') +  
 theme\_light()

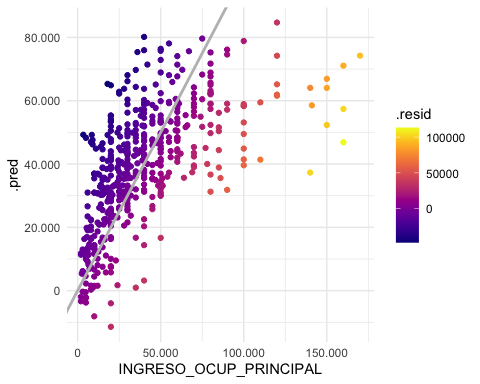


Con una muestra de mismo tamaño, graficamos los

datos\_muestra <- augment(lm\_fit, new\_data = data\_eph\_lm\_ajusted) %>%  
 select(INGRESO\_OCUP\_PRINCIPAL, .pred) %>%  
 sample\_n(size = sample\_size, replace = FALSE)   
  
ggplot(datos\_muestra, aes(x = INGRESO\_OCUP\_PRINCIPAL, y = .pred)) +  
 geom\_point() +  
 theme\_minimal() +  
 geom\_abline(  
 intercept = 0,  
 slope = 1,  
 linewidth = 1,  
 color = "grey"  
 ) +  
 scale\_color\_viridis(option = "C") +  
 scale\_y\_continuous(labels = comma\_format(big.mark = ".", decimal.mark = ",")) +  
 scale\_x\_continuous(labels = comma\_format(big.mark = ".", decimal.mark = ","))



datos\_muestra <- augment(lm\_fit, new\_data = data\_eph\_lm\_ajusted) %>%  
 sample\_n(size = sample\_size, replace = FALSE) %>%  
 mutate(.resid = INGRESO\_OCUP\_PRINCIPAL - .pred) %>%  
 dplyr::select(INGRESO\_OCUP\_PRINCIPAL, .pred, .resid)  
  
ggplot(datos\_muestra, aes(y = .pred, x = INGRESO\_OCUP\_PRINCIPAL, color = .resid)) +  
 geom\_point() +  
 theme\_minimal() +  
 geom\_abline(  
 intercept = 0,  
 slope = 1,  
 linewidth = 1,  
 color = "grey"  
 ) +  
 scale\_color\_viridis(option = "C") +  
 scale\_y\_continuous(labels = comma\_format(big.mark = ".", decimal.mark = ",")) +  
 scale\_x\_continuous(labels = comma\_format(big.mark = ".", decimal.mark = ","))



predicciones <- predict(lm\_fit, new\_data = data\_eph\_lm\_ajusted)  
  
resultados <- tibble(  
 Observado = data\_eph\_lm\_ajusted$INGRESO\_OCUP\_PRINCIPAL,  
 Predicho = predicciones$.pred  
)  
  
# Calcular métricas de evaluación del modelo  
rmse <- sqrt(mean((resultados$Observado - resultados$Predicho)^2))  
mae <- mean(abs(resultados$Observado - resultados$Predicho))  
r\_squared <- cor(resultados$Observado, resultados$Predicho)^2  
  
# Imprimir las métricas de evaluación  
cat("RMSE:", rmse, "\n")

## RMSE: 22124.08

cat("MAE:", mae, "\n")

## MAE: 15977.8

cat("R cuadrado:", r\_squared, "\n")

## R cuadrado: 0.4001873

-> El modelo ha logrado explicar de forma moderada el Ingreso por Ocupación Principal

## Modelo de Bugging

### Split y preprocesamiento

Trabajaremos con el modelo de Random Forest. - En primer lugar, hacemos un train - test split al 70% de la muestra. - Utilizamos la función *tune()* para el tuneo de hiperparámetros. - Luego, armamos un recipe para preprocesamiento, que convierte a dummy la variable *SEXO* y convierte a indicadores el resto de las categóricas no binarias.

library(recipes)  
set.seed(123)  
data\_split <- initial\_split(data\_eph, prop = 0.7)  
data\_train <- training(data\_split)  
data\_test <- testing(data\_split)  
  
  
tree\_spec <- bag\_tree(  
 cost\_complexity = tune(),  
 tree\_depth = tune(),  
 min\_n = tune()  
 ) %>%  
 set\_engine("rpart", times = 25) %>%  
 set\_mode("regression")  
  
tree\_spec %>% translate()

## Bagged Decision Tree Model Specification (regression)  
##   
## Main Arguments:  
## cost\_complexity = tune()  
## tree\_depth = tune()  
## min\_n = tune()  
##   
## Engine-Specific Arguments:  
## times = 25  
##   
## Computational engine: rpart   
##   
## Model fit template:  
## baguette::bagger(formula = missing\_arg(), data = missing\_arg(),   
## weights = missing\_arg(), cp = tune(), maxdepth = tune(),   
## minsplit = tune(), times = 25, base\_model = "CART")

recipe <- recipe(INGRESO\_OCUP\_PRINCIPAL ~ ., data = data\_train) %>%  
 #step\_normalize(all\_nominal()) %>%  
 step\_integer(all\_nominal(), -SEXO) %>%  
 step\_dummy(SEXO)  
  
recipe %>%   
 prep() %>%   
 juice() %>%   
 head(2)

## # A tibble: 2 × 13  
## AGLOMERADO NIVEL\_ED ROL\_HOGAR EDAD ESTADO\_CONYUGAL SECTOR\_PUB\_PRIV CAT\_OCUP  
## <int> <int> <int> <dbl> <int> <int> <int>  
## 1 6 7 3 22 3 3 2  
## 2 21 6 4 42 1 2 2  
## # ℹ 6 more variables: INTENSI <int>, CANT\_HORAS <dbl>, CATEGORIA <int>,  
## # CALIFICACION <int>, INGRESO\_OCUP\_PRINCIPAL <dbl>, SEXO\_Varon <dbl>

Trabajamos con el siguiente chunk:

tree\_bag <- workflow() %>% add\_model(tree\_spec)%>% add\_recipe(recipe)

tree\_grid <- grid\_regular(cost\_complexity(), tree\_depth(), min\_n(), levels = 4)

folds <- vfold\_cv(data\_train, v = 10) tree\_rs <- tree\_bag %>% tune\_grid( resamples = folds, grid = tree\_grid)

La optimización anterior nos ha resultado en un error (aparentemente por límite computacional / de recursos). Por ello, a pesar de que los pasos siguientes requerirían evaluar los resultados del cross validation pasaremos a los resultados del modelo simple sin optimización.

tree\_spec <- bag\_tree() %>%  
 set\_engine("rpart", times = 25) %>%  
 set\_mode("regression")  
  
tree\_bag <- workflow() %>%  
 add\_model(tree\_spec)%>%  
 add\_recipe(recipe)  
  
tree\_fit <- tree\_bag %>%  
 fit(data = data\_train)  
  
# Realizamos predicciones en el conjunto de prueba  
predictions <- tree\_fit %>%  
 predict(new\_data = data\_test)

extract\_fit\_parsnip(tree\_fit)

## parsnip model object  
##   
## Bagged CART (regression with 25 members)  
##   
## Variable importance scores include:  
##   
## # A tibble: 12 × 4  
## term value std.error used  
## <chr> <dbl> <dbl> <int>  
## 1 AGLOMERADO 1.17e13 1.40e12 25  
## 2 EDAD 9.83e12 1.06e12 25  
## 3 CANT\_HORAS 6.96e12 6.73e11 25  
## 4 CATEGORIA 4.71e12 2.56e11 25  
## 5 NIVEL\_ED 2.47e12 2.04e11 25  
## 6 CAT\_OCUP 2.40e12 6.06e11 25  
## 7 ESTADO\_CONYUGAL 2.31e12 3.60e11 25  
## 8 ROL\_HOGAR 2.06e12 5.26e11 25  
## 9 CALIFICACION 1.82e12 3.67e10 25  
## 10 INTENSI 1.62e12 2.03e11 25  
## 11 SEXO\_Varon 9.82e11 1.05e11 25  
## 12 SECTOR\_PUB\_PRIV 9.59e11 4.17e10 25

test <- tree\_fit %>%  
 predict(data\_test) %>%  
 bind\_cols(data\_test, .)  
  
metrics <- test %>%  
 metrics(truth = INGRESO\_OCUP\_PRINCIPAL, estimate = .pred)  
  
metrics

## # A tibble: 3 × 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 36698.   
## 2 rsq standard 0.172  
## 3 mae standard 20650.

-> Lógicamente, las métricas del primer modelo representan mayor fiabilidad a los datos reales. La causa principal que hallamos es que en este segundo no logramos llevar a cabo la optimización planteada.

En su capacidad de predicción como modelos de regresión, elegimos el primer modelo por sobre este segundo.

Notamos que a partir de los modelos planteados es dificil establecer una relación lineal entre las variables presentadas como explicativas y el Ingreso por Ocupación Principal pero a su vez el cierto nivel de predicción logrado, sobre todo en el primer modelo de regresión lineal múltiple, nos indica que es posible establecer una relación entre nuestra variable a explicar y el resto del set de datos.