Proyecto Final: Avance 3 Solución

Carlos Tabares

2022

El objetivo de este avance es estimar un modelo de Causal Machine Learning para poder predecir el impacto de otorgar un descuento a nivel cliente. Es decir, la demanda incremental en las compras derivada de la promoción. Los resultados del modelo se utilizarán para implementar una estrategia de descuentos focalizada que maximice la rentabilidad.

Instalación de librerías

```
required_pkgs <- c('tidyverse', 'dplyr','RCT','grf',"fastDummies")

installed_pkgs <- installed.packages()

missing_pkgs <- required_pkgs[!(required_pkgs %in% installed_pkgs[, 1])]

if (length(missing_pkgs) == 0 ) {
  message("Librerias cargadas")
} else {
  install.packages(missing_pkgs)

  message("Instalacion completa")

}
rm(installed_pkgs,missing_pkgs)

invisible(lapply(required_pkgs, library, character.only = TRUE))

rm(required_pkgs)
```

Pregunta 1

Explora la base y asegúrate de tener todas tus variables en formato numérico. Si tienes variables de texto o factores, transfórmalas a variables categóricas. Si tienes variables con valores vacíos decide si debes excluir a esas observaciones o mantenerlas y justifica tu decisión.

```
#Cargar la base
load("Bases output/inactivos_evaluacion.RData")

#Explorar la base
#glimpse(inactivos_db)
```

```r
#Generar la comparativa de medias
inactivos_db <- inactivos_db %>%
  mutate(grupo_edad = ntile_label(edad,4,0))%>%
  dummy_cols(select_columns = c("genero","dispositivo","canal_marketing",
                                "productos_interes","tipo_producto",
                                "localidad","grupo_edad"),
             ignore_na = T, remove_selected_columns =  T) %>%
  mutate_at(vars(c(starts_with(c("genero","dispositivo","canal_marketing",
                                 "productos_interes","tipo_producto","localidad")))),
            function(x) x = if_else(is.na(x),0,as.double(x)))%>%
    select(-c(email, edad, strata, missfit))


# Quitar caracteres especiales del nombre de las columnas
names(inactivos_db) <- make.names(names(inactivos_db))


# Analizar la distribucion de las variables
summary_stat <- summary_statistics(inactivos_db %>%
                                     select(-c(numero_cliente, treat, treatment)))

# Winzorizar las variables con outliers
inactivos_db <- inactivos_db%>%
  mutate_at(vars(monto_compra, valor_carrito,visitas_web,login_app),
            function(x) x = if_else(x > quantile(x, probs = 0.99, na.rm = T),
                                    quantile(x, probs = 0.99, na.rm = T), x))


# Valores faltantes
missings<-map_dbl(inactivos_db %>% select_all(),
                  ~100*sum(is.na(.))/nrow(inactivos_db))

missings[missings>0]
```

```
## named numeric(0)
```

Pregunta 2:

Estima una matriz de correlaciones de todas tus variables. Muestra los pares de variables que tienen más de 95% de correlación y elimina una de cada par multicolineal.

```r
# Construir la matriz de correlación
cor_matrix <- cor(inactivos_db %>%
                    select(-c(numero_cliente, treat, treatment)))

cor_matrix[upper.tri(cor_matrix, diag = T)] = NA

cor_tibble <- tibble(row = rep(rownames(cor_matrix), ncol(cor_matrix)),
                     col = rep(colnames(cor_matrix), each = ncol(cor_matrix)),
                     cor = as.vector(cor_matrix))

cor_tibble <- cor_tibble%>%filter(!is.na(cor))
```

```r
large_cor_tibble <- cor_tibble%>%filter(abs(cor)>=0.95)

# Eliminar variables altamente correlacionadas
inactivos_db <- inactivos_db%>%select(-all_of(large_cor_tibble$col))

# Guardar la base de estimacion
save(inactivos_db, file = "Bases output/inactivos_estimacion.RData")
```

Pregunta 3.

Quédate únicamente con tus clientes del grupo de control y el tratamiento con el descuento. Divide aleatoriamente a la población en 2 muestras: la muestra de entrenamiento (70% de las observaciones) y la muestra de validación (30% de las observaciones).

```r
inactivos_db <- inactivos_db%>%
  filter(treat != 1)

inactivos_db <- inactivos_db%>%
  mutate(training_set = rbinom(n = nrow(inactivos_db),1,0.7))

inactivos_training <- inactivos_db %>% filter(training_set==1)
```

Pregunta 4:

Estima un causal forest en la base de entrenamiento (Estima 1000 árboles)

```r
## Crear el set de covariables
X <- inactivos_training%>%
  select(-c(numero_cliente, treat, treatment, compra,
            valor_compra, training_set))

X <- as.matrix(X)

## Generar el vector del grupo de tratamiento
treat <- inactivos_training$treat

valor_compra <- inactivos_training$valor_compra

t0 <- Sys.time()

causal_hte <- causal_forest(X = X, Y = valor_compra, W = treat, num.trees = 3000)

t1 <- Sys.time()

t1 -  t0
```

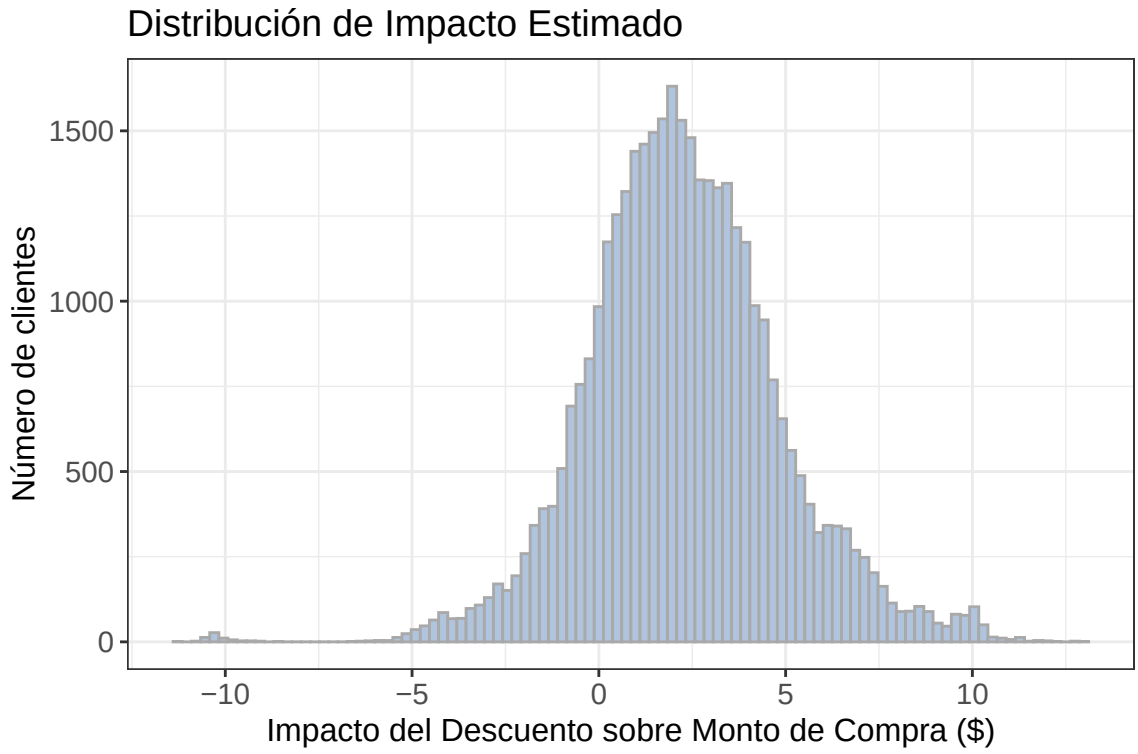
```
## Time difference of 2.599066 mins
```

```r
# Guardar el modelo
save(causal_hte, file = "Bases output/modelo_causal_forest.RData")
```

Pregunta 5:

Realiza un histograma para mostrar la distribución del impacto de tratamiento ¿Cuál es el impacto promedio de tratamiento estimado por el modelo? ¿Se asimila al efecto encontrado en el experimento?

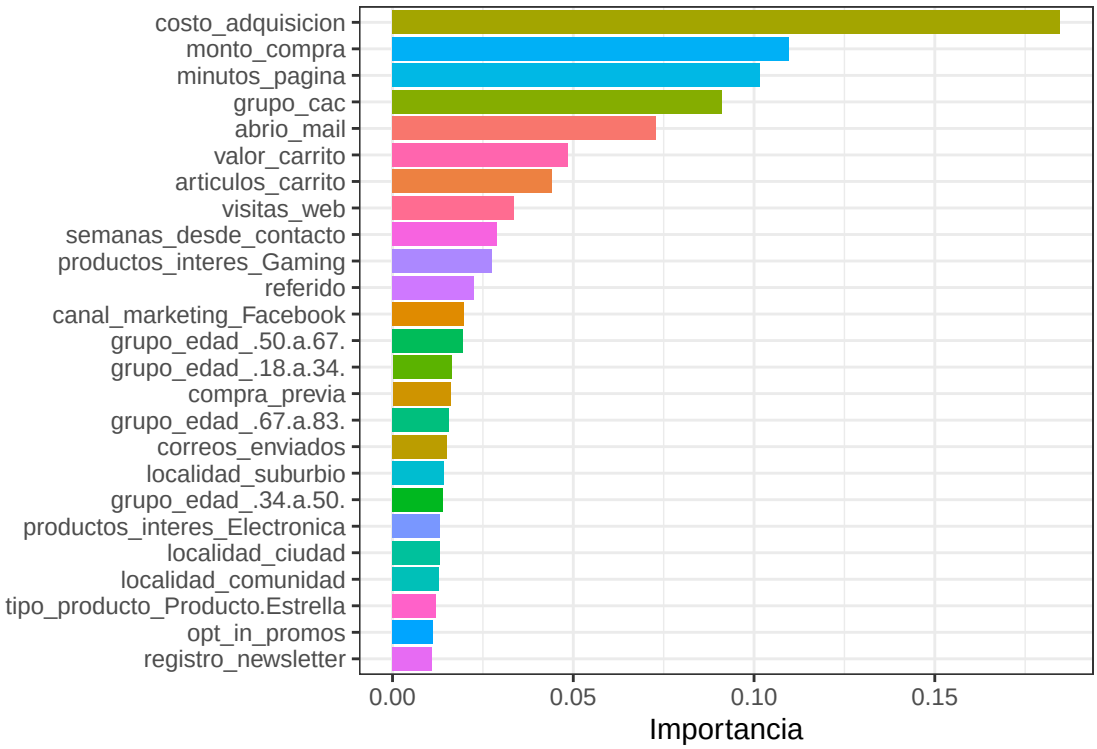
```
tau_in_sample = predict(causal_hte, estimate.variance = TRUE)

inactivos_training <- bind_cols(inactivos_training, tau_in_sample)

rm(tau_in_sample)

ggplot(inactivos_training)+
  geom_histogram(aes(predictions),bins = 100,
                 fill = 'lightsteelblue', color = 'darkgrey')+
  theme_bw()+
  labs(title = "Distribución de Impacto Estimado",
       x = 'Impacto del Descuento sobre Monto de Compra ($)',  y = 'Número de clientes')+
  theme(axis.text = element_text(size = 10.5),
        text = element_text(size = 12), legend.position = 'bottom')
```




```
## Predicción del efecto promedio de tratamiento
average_treatment_effect(causal_hte)
```

```
## estimate  std.err
## 2.279108 0.201566
```

```r
# Analizar la importancia de las variables en el criterio de particion de los arboles
var_importance<-variable_importance(causal_hte)
var_importance<-as.data.frame(var_importance)

var_importance<-
  var_importance %>%
  mutate(variable = colnames(X)) %>%
  rename(Importancia = V1)

write.table(var_importance, file ="clipboard", sep = "\t", row.names = F)


# Graficar la importancia de las variables
ggplot(var_importance %>% filter(Importancia>0.01))+
  geom_col(aes(fct_reorder(variable, Importancia), Importancia,fill = variable))+
  coord_flip()+theme_bw()+theme(legend.position = "none")+labs(x = "")
```




```r
rm(cor_matrix, cor_tibble, summary_stat, large_cor_tibble, X,
   var_importance, treat, valor_compra, missings)
```

Pregunta 6:

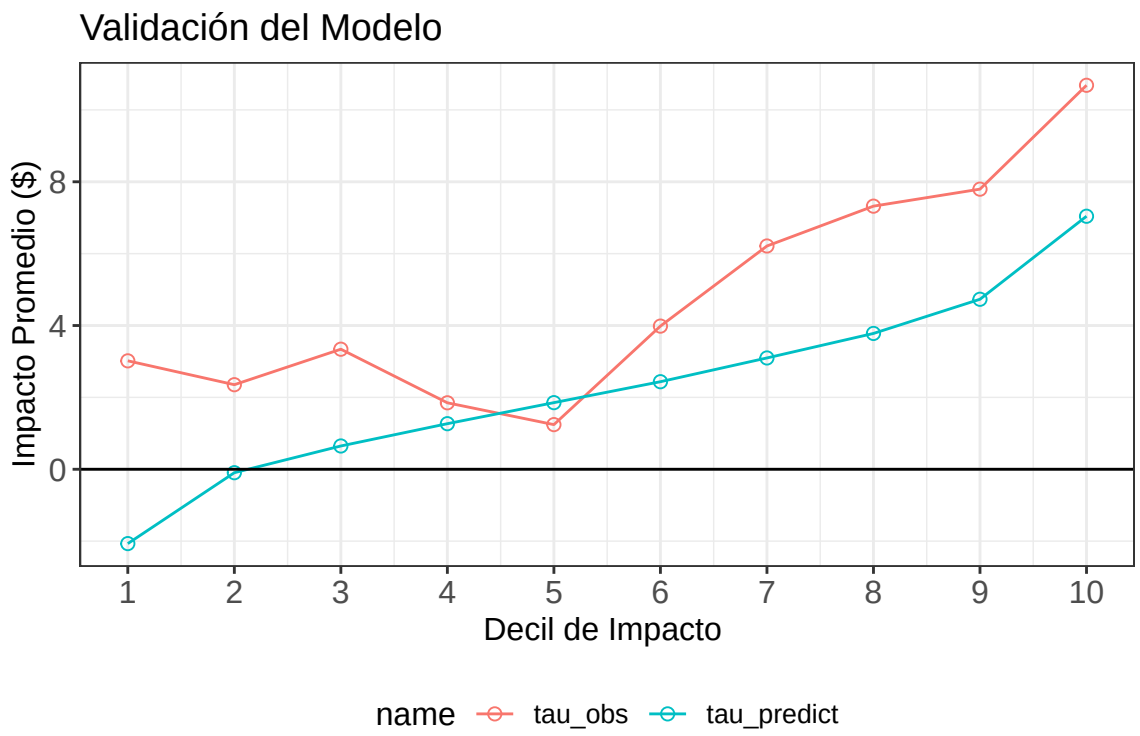
Evalua el poder predictivo del modelo en la base validacion. Recuerda dividir tu base de validación en k partes (k=10) con base en el score de predicción modelo. Posteriormente, estima el impacto de tratamiento (del experimento) en cada grupo de score y también calcula el promedio de las predicciones en cada grupo. (Tip: Puedes estimar el impacto de tratamiento con la función *impact_eval* considerando efectos heterogéneos por grupo de score). Valida si para los distintos grupos de score, la predicción del impacto promedio y el coeficiente de la regresión son crecientes y consistentes.

```r
inactivos_validation <- inactivos_db %>% filter(training_set==0)

## Creamos el set de covariables de la validacion
X <- inactivos_validation%>%
  select(-c(numero_cliente, treat, treatment, compra,
            valor_compra, training_set))

X <- as.matrix(X)

# Realizamos la prediccion
inactivos_validation <-  inactivos_validation %>%
  mutate(predictions = predict(causal_hte, newdata = X)$predictions)

summary(inactivos_validation$predictions)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -10.1717   0.6605   2.1426   2.2687   3.7735  12.2707
```

```r
#Crear los deciles por score predecido
inactivos_validation <-  inactivos_validation %>%
  mutate(score_group = as.integer(ntile(predictions, n = 10)))

# Checando poder de prediccion

ITT<-impact_eval(inactivos_validation,
                 endogenous_vars = "valor_compra",
                 treatment = "treat",
                 heterogenous_vars = "score_group")

ITT_score<-ITT$valor_compra_score_group

ITT_score <-  ITT_score %>% filter(term != "(Intercept)")

rm(ITT)

score_table <- inactivos_validation %>%
  group_by(score_group) %>%
  summarise(tau_predict = mean(predictions))

ITT_score<-
  left_join(ITT_score %>%
              select(score_group, estimate)%>%
              rename(tau_obs = estimate), score_table, by = "score_group")


ITT_score<-
  ITT_score %>%
  pivot_longer(cols = c(tau_predict, tau_obs))


ggplot(ITT_score, aes(x = score_group, y = value, color = name)) + geom_line() +
  geom_point(shape = 21, size = 2) +
  labs(title = "Validación del Modelo",
```

```
      x = "Decil de Impacto ", y="Impacto Promedio ($)")+
  theme_bw()+geom_hline(yintercept = 0)+
  theme(axis.text = element_text(size=12),
        text = element_text(size=12),legend.position = "bottom")+
  scale_x_continuous(breaks = seq(0, 10, 1))
```




```
rm(X, score_table, ITT_score, inactivos_training, inactivos_validation)
```

Pregunta 7:

Predice cuál hubiera sido el impacto sobre las ventas si los clientes que recibieron el cashback hubieran recibido un descuento. Asume que estos clientes nunca fueron tratados y úsalos para simular una estrategia de focalización a nivel usuario con base en los resultados de tu modelo. Asume que el monto de compra mínimo es de $7 y que sólo tienes presupuesto para dar 1000 cupones de descuento ¿Cuál es el impacto promedio y el impacto total esperado de los usuarios de tu campaña focalizada?

```
load("Bases output/inactivos_estimacion.RData")

inactivos_focalizacion <- inactivos_db %>%
  filter(treat==1)

# Generando el vector de covariables

X <- inactivos_focalizacion %>%
  select(-c(numero_cliente, treat,
            treatment, compra, valor_compra))
```

```r
# Revisar que todas las covariables del modelo estén en la matrix
variables_modelo<-colnames(causal_hte$X.orig)

en_ambas<-intersect(variables_modelo, names(X))

variables_faltantes<-setdiff(variables_modelo, names(X))

X <- as.matrix(X)

# Realizar la prediccion
inactivos_focalizacion <-  inactivos_focalizacion %>%
  mutate(predictions = predict(causal_hte, newdata = X)$predictions)

summary(inactivos_focalizacion$predictions)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -10.1717   0.6651   2.1604   2.2639   3.7845  12.2707
```

```r
## Filtrar primeros 1000 clientes más responsivos
inactivos_focalizacion <- inactivos_focalizacion%>%
  filter(rank(desc(predictions))<=1000 & predictions>7)


# Impacto promedio e impacto total
lift_table <- inactivos_focalizacion%>%
  group_by()%>%
  summarise(impacto_promedio = mean(predictions),
            impacto_esperado = impacto_promedio*1000*0.2080256)
```