Dominick’s Finer Foods

Duvan Santiago Castro - Miguel Ángel Malagón

2022-10-04

# Introducción

A lo largo de las siguientes secciones encontrará los diferentes modelos estimados y sus resultados correspondientes, que permitirá entender la demanda de cerveza de la cadena de restaurantes Dominick’s Finer Foods (DFF). Para dicha interpretación nos centraremos en 2 tipos de cervezas con sus respectivos precios.

Inicialmente realizaremos una serie de filtros que nos permitan seleccionar una muestra de 200 observaciones para el desarrollo de los modelos.

# Primeros pasos - Alistamiento datos

Para la extracción de los datos, vamos a utilizar la base de datos wber, provenientes de (DFF) y les aplicaremos los siguientes filtros:

* Store = 109: Nos centraremos en la tienda 109
* Precio > 0
* Unidades > 0
* Seleccionar 2 tipos de marcas:
  + BUDWEISER BEER
  + BUDWEISER LIGHT BEER

#intalacion de las librerías necesarias  
  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.7 ✔ stringr 1.4.0  
## ✔ tidyr 1.2.0 ✔ forcats 0.5.1  
## ✔ readr 2.1.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

datos\_cervezas <- read.csv(file.choose())

beer <- filter(datos\_cervezas,STORE==109,("MOVE">0 & "PRICE">0),(UPC==1820000016|UPC==1820000106))  
#set.seed(22)  
#beer <- beer[ sample(nrow(beer),size = 300,replace = FALSE),]

# Cambio los datos   
  
beer$UPC <- replace(beer$UPC, beer$UPC==1820000016, 0)  
beer$UPC <- replace(beer$UPC, beer$UPC==1820000106, 1)  
  
# Seleccionar las variables necesarias   
  
beer <- select(beer,unidades="MOVE",precio="PRICE",marca="UPC")  
head(beer)

## unidades precio marca  
## 1 18 3.49 0  
## 2 20 3.49 0  
## 3 24 3.49 0  
## 4 10 3.59 0  
## 5 26 3.59 0  
## 6 14 3.59 0

una vez realizados los filtros necesarios, hacemos un cambio los tipos de datos con el fin de establecer el siguiente modelo:

$$

Demanda=*{0}+*{1}precio+\_{2}marca\ marca:=\ :marca=0: marca=1:

$$

# Modelo - Estimación

Para la estimación del modelo planteado anteriormente vamos se define un vector para la demanda que representa la cantidad de unidades vendidas de cada producto, así mismo, se define una matriz de diseño que contiene, una columna de unos asociada al intercepto, otra columna de precios y tipo de marca definida como una variable Dummy.

En los siguientes apartados vamos a regresar diferentes modelos con el fin de revisar cual de los 3 ( Lin - Lin), (log -log) o (Log -Lig) se ajusta mejor a los datos.

## Modelo Lin - Lin

El primer modelo que vamos a revisar es un modelo lineal de forma matricial.

$$ Y=X+e\ Y= X=\ \

$$ Una vez realizada la estimación, los coeficientes asociados al modelo principal previamente mencionado, son los siguientes:

y <- as.matrix(beer$unidades)  
x <- as.matrix(cbind(1,beer$precio,beer$marca))  
colnames(x) <-c("b0","precio","marca")   
betas <- (solve(t(x)%\*%x)%\*%t(x)%\*%y)  
betas

## [,1]  
## b0 7.3818174  
## precio 0.5661266  
## marca -3.2327097

quedando de esta manera el siguiente modelo:

$$ =59,33-12,90*Precio-3.16*Marca

$$ *Análisis:* - La demanda de Cervezas de la tienda 109 cuyas variables precio y marca, muestran una serie de comportamientos donde por un aumento el precio de 1und, la demanda disminuye 12,90 y cuando es 1 es decir “Budweiser beer”, la demanda se reduce un -3,16. de otro modo es cero.

## Modelo Log - Lin

Ahora vamos a plantear un modelo Log-Lin y revisaremos los coeficientes asociados a cada variable explicativa de la demnada de cervezas de la tienda 109 de (DFF)

$$ ln(consumo)=*{0}+*{1}precio+\_{2}marca+e\

$$ Dado que estamos trabajando las regresiones de forma matricial, antes de correr la regresión la variable consumo, representada en unidades vendidas se le hace una tranformación logaritmica y luego si se corre la respectiva regresión.

y <- as.matrix(beer$unidades)  
x <- as.matrix(cbind(1,beer$precio,beer$marca))  
colnames(x) <-c("b0","precio","marca")   
lny <- log(y)  
betas2 <- (solve(t(x)%\*%x)%\*%t(x)%\*%lny)  
betas2

## [,1]  
## b0 -Inf  
## precio Inf  
## marca NaN

interpretación:

## Modelo Log - Log

Ahora vamos a plantear un modelo Log-Log en donde nos aplicaremos una transformación logaritmica a los precios, pero no la a la variable dummmy dado que el log(0) no existe, por lo tanto sera un modelo log-log en el par de conusmo y precio, como se muestra en el siguiente planteamiento.

$$ ln(consumo)=*{0}+*{1}ln(precio)+\_{2}marca+e\

$$
Dado que las regresiones se estan trabajando de manera matricial, la matriz de diseño X y la matriz de Y "Consumo" se trabajaria de la siguiente manera:
$$

X=

, Y=

$$ Una vez realizamos la regresión, nos da los siguientes coeficientes asociados a cada variable.

y <- as.matrix(beer$unidades)  
logx <- as.matrix(cbind(1,log(beer$precio),beer$marca))  
colnames(x) <-c("b0","precio","marca")   
lny <- log(y)  
  
betas3 <- (solve(t(logx)%\*%logx)%\*%t(logx)%\*%lny)  
betas3

## [,1]  
## [1,] NaN  
## [2,] NaN  
## [3,] NaN

Interpretación:

# Matriz de Varianza Covarianza para los estimadores

Teniendo en cuenta lo anterior, notamos que hay diferentes interpretaciones para cada modelo, pero seria conveniente también conocer la variación de dichos coeficientes para cada modelo, por esta razón vamos a calcular la matriz de varianzas-covarianzas para los diferentes estimadores de cada modelo, para eso utilizaremos la siguiente formula:

$$ =2(XtX)\ \ ^2= T=observaciones,K=parámetros

$$ Teniendo en cuenta que la estimación de la varianza se realiza apartir de los errores estimados, en cada modelo vamos a realizar dicho procedimiento para ajustarlo al tipo de modelo.

## Modelo Lin - Lin

Antes de calcular la matriz de varianza covarianza vamos a estimar los errores

y <- as.matrix(beer$unidades)  
x <- as.matrix(cbind(1,beer$precio,beer$marca))  
colnames(x) <-c("b0","precio","marca")   
betas <- (solve(t(x)%\*%x)%\*%t(x)%\*%y)  
y\_gorro <- x%\*%betas  
e\_gorro1 <- y-y\_gorro  
  
# Calcular sigma^2  
t <- as.numeric(nrow(y))  
k <- as.numeric(ncol(x))  
var1 <- as.numeric((t(e\_gorro1)%\*%e\_gorro1)/(t-k))  
var1

## [1] 20.45947

Una vez que tenemos calcula la varianza estimada, vamos a calcular nuestra matriz var\_cov

x <- as.matrix(cbind(1,beer$precio,beer$marca))  
var\_cov1 <- var1\*(solve(t(x)%\*%x))  
var\_cov1

## [,1] [,2] [,3]  
## [1,] 2.4321841 -0.61286918 -0.13456249  
## [2,] -0.6128692 0.16054345 0.01099832  
## [3,] -0.1345625 0.01099832 0.18590705

## Modelo Log - Lin

Ahora tenermos que tener encuenta que tipo de modelo estamos estimando para así poder contruir nuestra matriz de var-cov, en donde a diferencia del anterior modelo, tenemos que hacer la siguiente modificación:

$$
ln(consumo)=\beta\_{0}+\beta\_{1}precio+\beta\_{2}marca\\
\widehat{ln(consumo)}=X\hat{\beta} \quad \implies
\widehat{consumo}=e^{\widehat{ln(consumo)}}
$$

y <- as.matrix(beer$unidades)  
x <- as.matrix(cbind(1,beer$precio,beer$marca))  
colnames(x) <-c("b0","precio","marca")   
lny <- log(y)  
betas2 <- (solve(t(x)%\*%x)%\*%t(x)%\*%lny)  
  
log\_y\_gorro2 <- x%\*%betas  
y\_gorro2 <- exp(log\_y\_gorro2)  
e\_gorro2 <- y-y\_gorro2  
  
# Calcular sigma^2  
t <- as.numeric(nrow(y))  
k <- as.numeric(ncol(x))  
var2 <- as.numeric((t(e\_gorro2)%\*%e\_gorro2)/(t-k))  
var2

## [1] 102605119

x <- as.matrix(cbind(1,beer$precio,beer$marca))  
var\_cov2 <- var2\*(solve(t(x)%\*%x))  
var\_cov2

## [,1] [,2] [,3]  
## [1,] 12197506.6 -3073564.98 -674836.62  
## [2,] -3073565.0 805132.23 55157.02  
## [3,] -674836.6 55157.02 932331.75

## Modelo Log - Log

# R ajustado de cada especificación

## Modelo Lin - Lin

## Modelo Log - Lin

## Modelo Log - Log

# Teorema de Frisch-Waugh.

# Distancia de Cook

# Pronóstico