Predicción Ventas para mes de febrero

Cristopher Valenzuela

2022-06-05

En esta prueba se les entregará un dataset con 4 años de historia de venta de cierto tipo de bebida individual, para un local ubicado en el litoral central, el que contiene las siguientes columnas:

● ds: abreviatura de datestamp, fecha del dato

● precio\_ref: precio de referencia de la bebida ese día

● unidades\_total: total de unidades de bebida vendidas ese día

● monto\_total: monto total recaudado por estas bebidas ese día

El objetivo es generar un pronóstico de ventas diaria para el mes de Febrero 2022, para eso se cuenta con información que va desde Enero 2017 a Enero 2022, la que está en el archivo dataset\_ts\_ventas\_train.csv, mientras que las ventas reales de Febrero 2022 se encuentran en el archivo dataset\_ts\_ventas\_test.csv. Sobre estos datos puede incorporar supuestos e información adicional de dominio público para efectos de generar el pronóstico.

library(readr)  
data\_train <- read\_csv("C:/Users/crisv/Downloads/Desafo arcoprime/Prueba\_Técnica\_-\_Proceso\_Data\_Scientist/dataset\_ts\_ventas\_train.csv")

## Rows: 1854 Columns: 4  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (3): precio\_ref, unidades\_total, monto\_total  
## date (1): ds  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

data\_train$ds = as.Date(data\_train$ds)  
  
head(data\_train)

## # A tibble: 6 × 4  
## ds precio\_ref unidades\_total monto\_total  
## <date> <dbl> <dbl> <dbl>  
## 1 2017-01-01 990 158 142740  
## 2 2017-01-02 990 152 137180  
## 3 2017-01-03 990 125 115390  
## 4 2017-01-04 990 107 98710  
## 5 2017-01-05 990 103 96270  
## 6 2017-01-06 990 93 84808

str(data\_train)

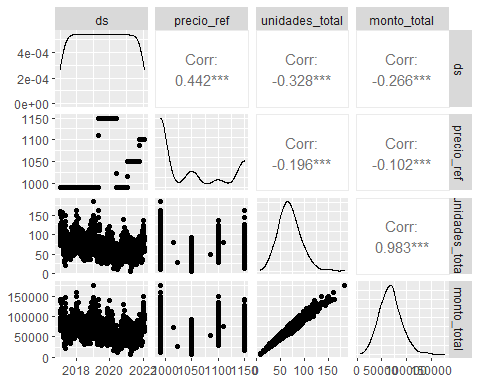
## spec\_tbl\_df [1,854 × 4] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ ds : Date[1:1854], format: "2017-01-01" "2017-01-02" ...  
## $ precio\_ref : num [1:1854] 990 990 990 990 990 990 990 990 990 990 ...  
## $ unidades\_total: num [1:1854] 158 152 125 107 103 93 156 96 95 106 ...  
## $ monto\_total : num [1:1854] 142740 137180 115390 98710 96270 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. ds = col\_date(format = ""),  
## .. precio\_ref = col\_double(),  
## .. unidades\_total = col\_double(),  
## .. monto\_total = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

library(GGally)

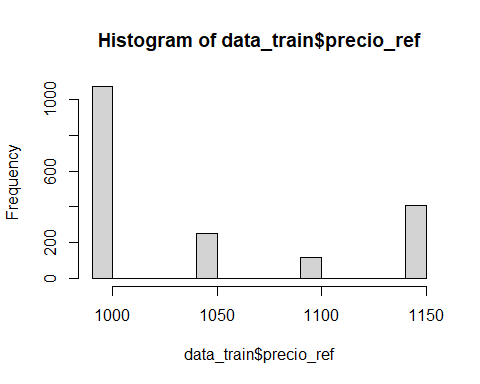
## Loading required package: ggplot2

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

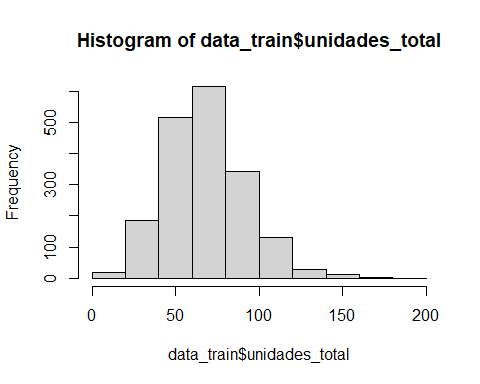
GGally::ggpairs(data\_train)



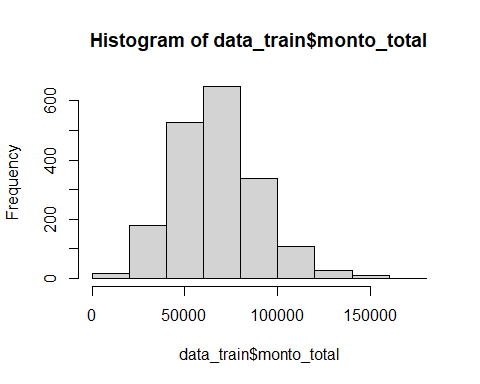
hist(data\_train$precio\_ref)



hist(data\_train$unidades\_total)



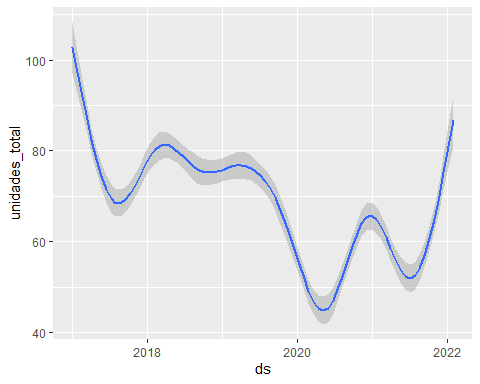
hist(data\_train$monto\_total)



Dos variables parecen tener distribucion normal, y parecen tener una alta correlacion entre si, ademas parecen ser estacionarias, una de las variable como el precio no parece tener una variabilidad suficiente para ser tratada como una serie temporal.

data\_train %>%  
 ggplot(aes(ds,unidades\_total)) +  
 geom\_smooth()

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



Utilizamos lubridate para agrupar los dias

library(lubridate)

##   
## Attaching package: 'lubridate'

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

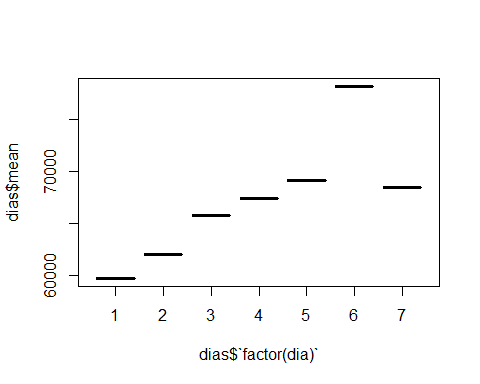
library(tidyverse)

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

## ✔ tibble 3.1.7 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ purrr 0.3.4 ✔ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ lubridate::as.difftime() masks base::as.difftime()  
## ✖ lubridate::date() masks base::date()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ lubridate::intersect() masks base::intersect()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ lubridate::setdiff() masks base::setdiff()  
## ✖ lubridate::union() masks base::union()

data\_train2 = data\_train  
  
  
data\_train2$dia = wday(data\_train2$ds)  
  
dias =data\_train2 %>%  
 group\_by(factor(dia)) %>%  
 summarize(mean = mean(monto\_total))  
  
  
boxplot(dias$mean~dias$`factor(dia)`)



El boxplot muestras que el dia 6 de la semana tiene las mayores ventas promedio.Mientras que las ventas mas bajas parecen ser de los dias 1 y 2.

#Podemos hacer una regresion lineal como variable dependiente las ventas y el dia como independiente  
  
summary(lm(data\_train2$monto\_total~as.factor(data\_train2$dia),data = data\_train2))

##   
## Call:  
## lm(formula = data\_train2$monto\_total ~ as.factor(data\_train2$dia),   
## data = data\_train2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -61685 -14147 -1033 12483 108934   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 59696 1354 44.104 < 2e-16 \*\*\*  
## as.factor(data\_train2$dia)2 2346 1912 1.227 0.22004   
## as.factor(data\_train2$dia)3 6090 1914 3.182 0.00149 \*\*   
## as.factor(data\_train2$dia)4 7643 1916 3.989 6.90e-05 \*\*\*  
## as.factor(data\_train2$dia)5 9412 1914 4.917 9.56e-07 \*\*\*  
## as.factor(data\_train2$dia)6 18474 1914 9.651 < 2e-16 \*\*\*  
## as.factor(data\_train2$dia)7 8723 1916 4.553 5.64e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22030 on 1847 degrees of freedom  
## Multiple R-squared: 0.05856, Adjusted R-squared: 0.0555   
## F-statistic: 19.15 on 6 and 1847 DF, p-value: < 2.2e-16

Existen relaciones estadisticamente significativas entre el dia y las ventas pero los dias explican una parte pequeña de las ventas, el ajuste del modelo es bajo 0.05.

cor(data\_train2$monto\_total,data\_train2$dia)

## [1] 0.1947814

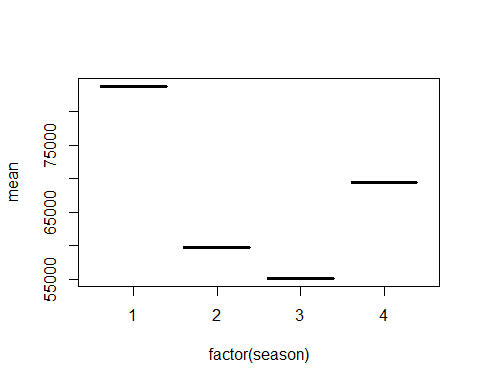
Existe una correlacion debil entre el dia de la semana y las ventas.

# Creamos una variable con el mes del año  
data\_train2$mes = month(data\_train2$ds)  
  
data\_train3 = data\_train2  
  
#Utilizamos la nueva variable que tiene el mes, para crear la epoca del año, asumiremos que son datos internacionales, asi que utilizamos las  
#estaciones del año internacionales  
#Invierno: diciembre - febrero = 1  
#Primavera: marzo - mayo = 2  
#Verano: junio - agosto = 3   
#Otoño: septiembre - noviembre = 4  
  
estaciones <- data\_train3 %>%   
 mutate(season =   
 ifelse(mes %in% c(12, 1, 2), "1",  
 ifelse(mes %in% c(3, 4, 5), "2",  
 ifelse(mes %in% c(6, 7, 8), "3",  
 ifelse(mes %in% c(9, 10, 11), "4", "Error")))))  
  
#realizamos una regresion con las estaciones como variables predictoras  
summary(lm(estaciones$monto\_total~as.factor(estaciones$season)))

##   
## Call:  
## lm(formula = estaciones$monto\_total ~ as.factor(estaciones$season))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -54122 -13249 -837 11766 92521   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 83752 902 92.85 <2e-16 \*\*\*  
## as.factor(estaciones$season)2 -23981 1293 -18.55 <2e-16 \*\*\*  
## as.factor(estaciones$season)3 -28654 1291 -22.20 <2e-16 \*\*\*  
## as.factor(estaciones$season)4 -14310 1294 -11.05 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 19800 on 1850 degrees of freedom  
## Multiple R-squared: 0.2382, Adjusted R-squared: 0.237   
## F-statistic: 192.9 on 3 and 1850 DF, p-value: < 2.2e-16

Existe una relacion estadisticamente significativa entre las estaciones y las ventas, las estaciones explican aproximadamente el 23% en la variacion de las ventas.

estaciones1 = estaciones %>%  
 group\_by(factor(season)) %>%  
 summarize(mean = mean(monto\_total))  
  
  
plot(estaciones1)



En este caso el periodo que va desde diciembre a febrero parece tener la mayor cantidad de ventas promedio, mientras que entre junio y agosto existe la menor cantidad de ventas promedio.

Generamos un periodo de tiempo que abarque el inicio de la pandemia, tomando como referencia marzo del 2020

estaciones3 = estaciones  
  
  
pandemia = rep(c(0,1),each=1155)  
pandemia = as.data.frame(pandemia)  
pandemia = pandemia[-c(1855:2310),]  
pandemia = as.data.frame(pandemia)  
  
estaciones3 = cbind(estaciones3,pandemia)  
head(estaciones3)

## ds precio\_ref unidades\_total monto\_total dia mes season pandemia  
## 1 2017-01-01 990 158 142740 1 1 1 0  
## 2 2017-01-02 990 152 137180 2 1 1 0  
## 3 2017-01-03 990 125 115390 3 1 1 0  
## 4 2017-01-04 990 107 98710 4 1 1 0  
## 5 2017-01-05 990 103 96270 5 1 1 0  
## 6 2017-01-06 990 93 84808 6 1 1 0

summary(lm(estaciones3$monto\_total~as.factor(estaciones3$pandemia),data = estaciones3))

##   
## Call:  
## lm(formula = estaciones3$monto\_total ~ as.factor(estaciones3$pandemia),   
## data = estaciones3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -52171 -14921 -1828 13161 102702   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 73571.1 622.5 118.19 <2e-16 \*\*\*  
## as.factor(estaciones3$pandemia)1 -16846.3 1013.8 -16.62 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 21160 on 1852 degrees of freedom  
## Multiple R-squared: 0.1297, Adjusted R-squared: 0.1293   
## F-statistic: 276.1 on 1 and 1852 DF, p-value: < 2.2e-16

cor(estaciones3$monto\_total,estaciones3$pandemia)

## [1] -0.3602019

El periodo de inicio de tiempo de la pandemia hasta los ultimos dias de los datos, parece indicar que la pandemia afecto negativamente de forma moderada las ventas. La pandemia explico aproximadamente el 12% de la variacion de los datos en las ventas.

Si nos preguntamos si las otras variables tienen efecto en las ventas

modelo<- lm(estaciones3$monto\_total~estaciones3$precio\_ref,data = estaciones3)  
summary(modelo)

##   
## Call:  
## lm(formula = estaciones3$monto\_total ~ estaciones3$precio\_ref,   
## data = estaciones3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -60152 -15203 -732 13661 107287   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 103630.787 8264.978 12.539 < 2e-16 \*\*\*  
## estaciones3$precio\_ref -34.995 7.928 -4.414 1.07e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22560 on 1852 degrees of freedom  
## Multiple R-squared: 0.01041, Adjusted R-squared: 0.009878   
## F-statistic: 19.49 on 1 and 1852 DF, p-value: 1.072e-05

cor(estaciones3$precio\_ref,estaciones3$monto\_total)

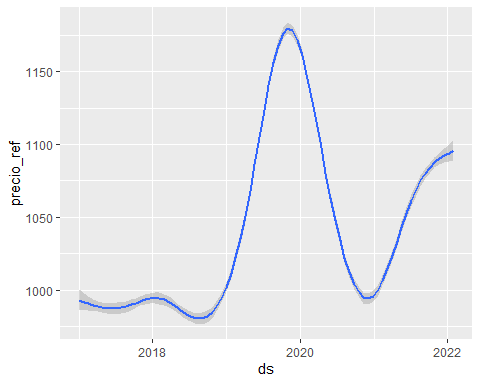
## [1] -0.1020407

Existe una relacion negativa debil entre el precio y las ventas, dicha relacion parece ser bastante debil en el modelo, el precio parece explicar el 1% de la variacion en las ventas.

Podriamos detectar los dias en que existio una promocion por el precio respectivo de los productos y sus ventas

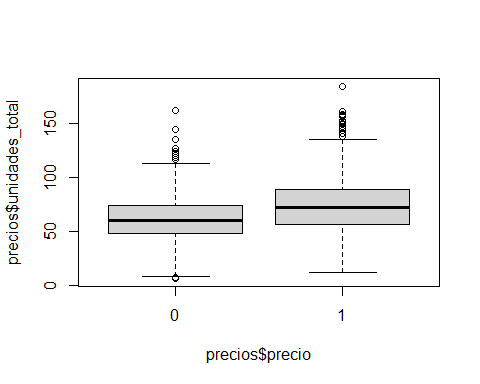
estaciones3 %>%  
 ggplot(aes(ds,precio\_ref)) +   
 geom\_smooth()

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



Parecen existir diferencias en los precios segun la fecha. Para detectar ofertas podriamos detectar valores fuera de la media o agrupaciones segun el tiempo. Podriamos comparar los precios por estacion

#Podriamos calcular los dias en los que el precio promedio es inferior.  
  
estaciones4 = estaciones3  
  
 precios <-estaciones4 %>%  
 group\_by(dia) %>%  
 mutate("precio" = precio\_ref < mean(precio\_ref))  
   
  
precios$precio = as.numeric(precios$precio)  
   
boxplot(precios$unidades\_total~precios$precio)



Los dias en que los precios estan por debajo la media historica, parecen tener algunos aumentos en las unidades vendidas, con intervalos y valores atipicos.

summary(lm(precios$unidades\_total~precios$precio))

##   
## Call:  
## lm(formula = precios$unidades\_total ~ precios$precio)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -61.515 -14.515 -1.449 14.485 110.485   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 61.4492 0.8305 73.99 <2e-16 \*\*\*  
## precios$precio 12.0662 1.0896 11.07 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 23.15 on 1852 degrees of freedom  
## Multiple R-squared: 0.0621, Adjusted R-squared: 0.06159   
## F-statistic: 122.6 on 1 and 1852 DF, p-value: < 2.2e-16

Existe una relacion estadisticamente entre las unidades totales y los precios bajo la media, pero el modelo explica menos del 1%.

summary(lm(precios$monto\_total~precios$precio))

##   
## Call:  
## lm(formula = precios$monto\_total ~ precios$precio)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -58382 -14822 -1160 13287 106011   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 63002.9 803.4 78.424 < 2e-16 \*\*\*  
## precios$precio 7259.0 1054.0 6.887 7.79e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 22390 on 1852 degrees of freedom  
## Multiple R-squared: 0.02497, Adjusted R-squared: 0.02444   
## F-statistic: 47.43 on 1 and 1852 DF, p-value: 7.788e-12

Existe una relacion estadisticamente entre el monto total y los precios bajo la media, pero el modelo explica menos del 1%.

Realizamos un pronostico de venta diaria

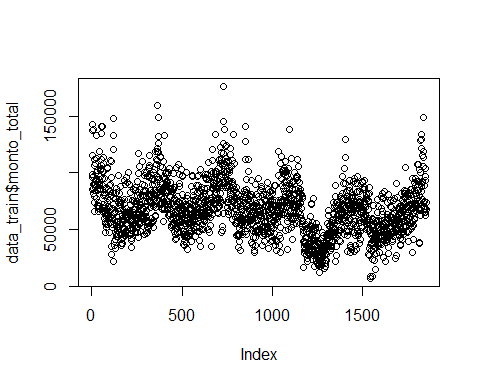
El primer paso es realizar un analisis exploratorio

Chequeamos la estacionariedad

library(tseries)

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

plot(data\_train$monto\_total)



Realizamos un analisis con ARIMA con variables exogenas.

########### Modelo VAR con variables exógenas  
#Variables endógenas  
#Variables endógenas  
dat.VAR <- cbind(precios$precio\_ref,precios$unidades\_total,precios$monto\_total,as.factor(precios$season),as.factor(precios$pandemia),as.factor(precios$precio))  
colnames(dat.VAR) <- c("precio\_ref","unidades\_total","monto\_total","season","pandemia","precio")  
  
  
dat.EXO <- cbind(as.factor(precios$season),as.factor(precios$pandemia),as.factor(precios$precio))  
colnames(dat.EXO) <- c("season","pandemia","precio")  
  
head(dat.EXO)

## season pandemia precio  
## [1,] 1 1 2  
## [2,] 1 1 2  
## [3,] 1 1 2  
## [4,] 1 1 2  
## [5,] 1 1 2  
## [6,] 1 1 2

VAR.est <- vars::VAR(dat.VAR,lag.max=28, ic="AIC",type="const")  
summary(VAR.est)

##   
## VAR Estimation Results:  
## =========================   
## Endogenous variables: precio\_ref, unidades\_total, monto\_total, season, pandemia, precio   
## Deterministic variables: const   
## Sample size: 1851   
## Log Likelihood: -20171.1   
## Roots of the characteristic polynomial:  
## 0.9989 0.9954 0.9954 0.9877 0.9465 0.7482 0.442 0.442 0.4098 0.4098 0.3293 0.3293 0.214 0.214 0.1124 0.08974 0.08974 0.02919  
## Call:  
## vars::VAR(y = dat.VAR, type = "const", lag.max = 28, ic = "AIC")  
##   
##   
## Estimation results for equation precio\_ref:   
## ===========================================   
## precio\_ref = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 1.422e+00 4.658e-02 30.528 < 2e-16 \*\*\*  
## unidades\_total.l1 4.829e-02 5.173e-02 0.933 0.35073   
## monto\_total.l1 -4.745e-05 5.327e-05 -0.891 0.37321   
## season.l1 -2.169e+00 5.514e-01 -3.933 8.72e-05 \*\*\*  
## pandemia.l1 -2.322e-02 4.291e+00 -0.005 0.99568   
## precio.l1 1.474e+01 4.998e+00 2.949 0.00323 \*\*   
## precio\_ref.l2 -5.361e-01 6.673e-02 -8.033 1.68e-15 \*\*\*  
## unidades\_total.l2 -2.740e-02 5.659e-02 -0.484 0.62828   
## monto\_total.l2 3.361e-05 5.830e-05 0.577 0.56433   
## season.l2 2.228e+00 7.803e-01 2.856 0.00434 \*\*   
## pandemia.l2 2.886e-01 6.032e+00 0.048 0.96184   
## precio.l2 -1.508e+01 6.236e+00 -2.419 0.01567 \*   
## precio\_ref.l3 1.069e-01 4.289e-02 2.493 0.01275 \*   
## unidades\_total.l3 -2.923e-02 5.169e-02 -0.566 0.57177   
## monto\_total.l3 2.336e-05 5.330e-05 0.438 0.66126   
## season.l3 -9.597e-02 5.573e-01 -0.172 0.86329   
## pandemia.l3 -5.493e-01 4.258e+00 -0.129 0.89735   
## precio.l3 -3.132e-01 4.752e+00 -0.066 0.94746   
## const 8.874e+00 5.363e+00 1.654 0.09820 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 4.245 on 1832 degrees of freedom  
## Multiple R-Squared: 0.9959, Adjusted R-squared: 0.9959   
## F-statistic: 2.484e+04 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
## Estimation results for equation unidades\_total:   
## ===============================================   
## unidades\_total = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 3.173e-01 1.926e-01 1.648 0.099618 .   
## unidades\_total.l1 2.242e-01 2.138e-01 1.049 0.294484   
## monto\_total.l1 2.374e-04 2.202e-04 1.078 0.281145   
## season.l1 -1.010e+00 2.279e+00 -0.443 0.657875   
## pandemia.l1 -1.731e+01 1.774e+01 -0.976 0.329269   
## precio.l1 3.845e+01 2.066e+01 1.861 0.062860 .   
## precio\_ref.l2 -3.765e-01 2.759e-01 -1.365 0.172466   
## unidades\_total.l2 3.369e-01 2.339e-01 1.440 0.149983   
## monto\_total.l2 -2.963e-04 2.410e-04 -1.230 0.219034   
## season.l2 -2.377e+00 3.225e+00 -0.737 0.461291   
## pandemia.l2 1.210e+01 2.493e+01 0.485 0.627566   
## precio.l2 -3.640e+01 2.578e+01 -1.412 0.158136   
## precio\_ref.l3 2.451e-02 1.773e-01 0.138 0.890054   
## unidades\_total.l3 2.948e-01 2.137e-01 1.380 0.167825   
## monto\_total.l3 -1.678e-04 2.203e-04 -0.762 0.446283   
## season.l3 1.843e+00 2.304e+00 0.800 0.423680   
## pandemia.l3 -1.547e+00 1.760e+01 -0.088 0.929964   
## precio.l3 -5.324e+00 1.964e+01 -0.271 0.786417   
## const 7.949e+01 2.217e+01 3.585 0.000346 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 17.55 on 1832 degrees of freedom  
## Multiple R-Squared: 0.4576, Adjusted R-squared: 0.4523   
## F-statistic: 85.87 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
## Estimation results for equation monto\_total:   
## ============================================   
## monto\_total = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 3.268e+02 1.871e+02 1.747 0.080864 .   
## unidades\_total.l1 -2.612e+02 2.078e+02 -1.257 0.208926   
## monto\_total.l1 7.217e-01 2.140e-01 3.373 0.000760 \*\*\*  
## season.l1 -1.116e+03 2.215e+03 -0.504 0.614430   
## pandemia.l1 -1.817e+04 1.723e+04 -1.054 0.291947   
## precio.l1 4.032e+04 2.007e+04 2.009 0.044715 \*   
## precio\_ref.l2 -4.072e+02 2.680e+02 -1.519 0.128939   
## unidades\_total.l2 1.023e+02 2.273e+02 0.450 0.652653   
## monto\_total.l2 -5.320e-02 2.342e-01 -0.227 0.820299   
## season.l2 -1.995e+03 3.134e+03 -0.637 0.524504   
## pandemia.l2 1.503e+04 2.423e+04 0.620 0.535008   
## precio.l2 -3.979e+04 2.505e+04 -1.588 0.112363   
## precio\_ref.l3 4.731e+01 1.723e+02 0.275 0.783620   
## unidades\_total.l3 1.034e+02 2.076e+02 0.498 0.618567   
## monto\_total.l3 2.275e-02 2.141e-01 0.106 0.915384   
## season.l3 1.581e+03 2.238e+03 0.707 0.479942   
## pandemia.l3 -3.525e+03 1.710e+04 -0.206 0.836728   
## precio.l3 -4.051e+03 1.909e+04 -0.212 0.831943   
## const 7.740e+04 2.154e+04 3.593 0.000336 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 17050 on 1832 degrees of freedom  
## Multiple R-Squared: 0.4332, Adjusted R-squared: 0.4276   
## F-statistic: 77.79 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
## Estimation results for equation season:   
## =======================================   
## season = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 2.103e-04 1.973e-03 0.107 0.915   
## unidades\_total.l1 -2.788e-03 2.191e-03 -1.273 0.203   
## monto\_total.l1 2.556e-06 2.256e-06 1.133 0.257   
## season.l1 9.878e-01 2.335e-02 42.307 <2e-16 \*\*\*  
## pandemia.l1 8.315e-04 1.817e-01 0.005 0.996   
## precio.l1 3.340e-02 2.116e-01 0.158 0.875   
## precio\_ref.l2 -1.081e-05 2.826e-03 -0.004 0.997   
## unidades\_total.l2 3.664e-04 2.396e-03 0.153 0.878   
## monto\_total.l2 -6.420e-07 2.469e-06 -0.260 0.795   
## season.l2 1.671e-03 3.304e-02 0.051 0.960   
## pandemia.l2 -6.823e-03 2.554e-01 -0.027 0.979   
## precio.l2 -4.073e-04 2.640e-01 -0.002 0.999   
## precio\_ref.l3 -3.971e-04 1.816e-03 -0.219 0.827   
## unidades\_total.l3 2.172e-03 2.189e-03 0.993 0.321   
## monto\_total.l3 -2.487e-06 2.257e-06 -1.102 0.271   
## season.l3 -6.207e-03 2.360e-02 -0.263 0.793   
## pandemia.l3 -1.465e-02 1.803e-01 -0.081 0.935   
## precio.l3 -5.909e-02 2.012e-01 -0.294 0.769   
## const 3.722e-01 2.271e-01 1.639 0.101   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 0.1797 on 1832 degrees of freedom  
## Multiple R-Squared: 0.9746, Adjusted R-squared: 0.9744   
## F-statistic: 3905 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
## Estimation results for equation pandemia:   
## =========================================   
## pandemia = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 2.662e-06 2.537e-04 0.010 0.992   
## unidades\_total.l1 2.803e-04 2.817e-04 0.995 0.320   
## monto\_total.l1 -2.767e-07 2.901e-07 -0.954 0.340   
## season.l1 1.651e-02 3.003e-03 5.497 4.40e-08 \*\*\*  
## pandemia.l1 9.972e-01 2.337e-02 42.679 < 2e-16 \*\*\*  
## precio.l1 -1.522e-03 2.721e-02 -0.056 0.955   
## precio\_ref.l2 -2.402e-06 3.634e-04 -0.007 0.995   
## unidades\_total.l2 -4.750e-04 3.081e-04 -1.542 0.123   
## monto\_total.l2 4.944e-07 3.175e-07 1.558 0.120   
## season.l2 -1.659e-02 4.249e-03 -3.906 9.73e-05 \*\*\*  
## pandemia.l2 -1.378e-05 3.284e-02 0.000 1.000   
## precio.l2 3.050e-03 3.396e-02 0.090 0.928   
## precio\_ref.l3 1.998e-05 2.335e-04 0.086 0.932   
## unidades\_total.l3 1.194e-04 2.815e-04 0.424 0.672   
## monto\_total.l3 -1.201e-07 2.902e-07 -0.414 0.679   
## season.l3 -2.201e-04 3.034e-03 -0.073 0.942   
## pandemia.l3 2.391e-03 2.318e-02 0.103 0.918   
## precio.l3 -3.022e-04 2.588e-02 -0.012 0.991   
## const -2.254e-02 2.921e-02 -0.772 0.440   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 0.02311 on 1832 degrees of freedom  
## Multiple R-Squared: 0.9978, Adjusted R-squared: 0.9977   
## F-statistic: 4.513e+04 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
## Estimation results for equation precio:   
## =======================================   
## precio = precio\_ref.l1 + unidades\_total.l1 + monto\_total.l1 + season.l1 + pandemia.l1 + precio.l1 + precio\_ref.l2 + unidades\_total.l2 + monto\_total.l2 + season.l2 + pandemia.l2 + precio.l2 + precio\_ref.l3 + unidades\_total.l3 + monto\_total.l3 + season.l3 + pandemia.l3 + precio.l3 + const   
##   
## Estimate Std. Error t value Pr(>|t|)   
## precio\_ref.l1 -3.114e-03 4.359e-04 -7.145 1.29e-12 \*\*\*  
## unidades\_total.l1 -6.512e-04 4.840e-04 -1.345 0.178700   
## monto\_total.l1 6.624e-07 4.985e-07 1.329 0.184044   
## season.l1 1.714e-02 5.159e-03 3.323 0.000908 \*\*\*  
## pandemia.l1 1.575e-03 4.015e-02 0.039 0.968715   
## precio.l1 7.104e-01 4.676e-02 15.193 < 2e-16 \*\*\*  
## precio\_ref.l2 3.439e-03 6.244e-04 5.508 4.14e-08 \*\*\*  
## unidades\_total.l2 3.960e-04 5.294e-04 0.748 0.454634   
## monto\_total.l2 -4.512e-07 5.455e-07 -0.827 0.408228   
## season.l2 -1.825e-02 7.300e-03 -2.499 0.012533 \*   
## pandemia.l2 -3.403e-03 5.643e-02 -0.060 0.951921   
## precio.l2 2.417e-01 5.835e-02 4.143 3.58e-05 \*\*\*  
## precio\_ref.l3 -3.098e-04 4.013e-04 -0.772 0.440239   
## unidades\_total.l3 2.591e-04 4.836e-04 0.536 0.592256   
## monto\_total.l3 -2.209e-07 4.987e-07 -0.443 0.657869   
## season.l3 2.159e-03 5.214e-03 0.414 0.678823   
## pandemia.l3 1.395e-03 3.984e-02 0.035 0.972073   
## precio.l3 4.641e-02 4.446e-02 1.044 0.296707   
## const -1.571e-02 5.018e-02 -0.313 0.754278   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
##   
## Residual standard error: 0.03972 on 1832 degrees of freedom  
## Multiple R-Squared: 0.9936, Adjusted R-squared: 0.9935   
## F-statistic: 1.578e+04 on 18 and 1832 DF, p-value: < 2.2e-16   
##   
##   
##   
## Covariance matrix of residuals:  
## precio\_ref unidades\_total monto\_total season pandemia  
## precio\_ref 1.802e+01 6.429e-01 1.067e+03 -2.080e-04 1.181e-03  
## unidades\_total 6.429e-01 3.079e+02 2.974e+05 1.234e-02 -2.776e-03  
## monto\_total 1.067e+03 2.974e+05 2.907e+08 1.485e+01 -3.059e+00  
## season -2.080e-04 1.234e-02 1.485e+01 3.231e-02 -2.456e-07  
## pandemia 1.181e-03 -2.776e-03 -3.059e+00 -2.456e-07 5.343e-04  
## precio -1.461e-01 3.789e-03 -1.397e+00 7.248e-06 -8.748e-06  
## precio  
## precio\_ref -1.461e-01  
## unidades\_total 3.789e-03  
## monto\_total -1.397e+00  
## season 7.248e-06  
## pandemia -8.748e-06  
## precio 1.577e-03  
##   
## Correlation matrix of residuals:  
## precio\_ref unidades\_total monto\_total season pandemia  
## precio\_ref 1.0000000 0.008631 0.014747 -2.726e-04 1.203e-02  
## unidades\_total 0.0086313 1.000000 0.994226 3.913e-03 -6.845e-03  
## monto\_total 0.0147467 0.994226 1.000000 4.846e-03 -7.763e-03  
## season -0.0002726 0.003913 0.004846 1.000e+00 -5.911e-05  
## pandemia 0.0120328 -0.006845 -0.007763 -5.911e-05 1.000e+00  
## precio -0.8665908 0.005437 -0.002064 1.015e-03 -9.529e-03  
## precio  
## precio\_ref -0.866591  
## unidades\_total 0.005437  
## monto\_total -0.002064  
## season 0.001015  
## pandemia -0.009529  
## precio 1.000000

library(forecast)  
(fit <- auto.arima(dat.VAR[,3],  
 xreg=dat.VAR[,-3]))

## Series: dat.VAR[, 3]   
## Regression with ARIMA(1,0,1) errors   
##   
## Coefficients:  
## ar1 ma1 precio\_ref unidades\_total season pandemia  
## 0.9691 -0.5065 4.2676 965.4136 180.0001 -129.120  
## s.e. 0.0070 0.0231 1.7882 2.3587 195.1602 1031.595  
## precio  
## -2208.5236  
## s.e. 702.1504  
##   
## sigma^2 = 3325251: log likelihood = -16548.89  
## AIC=33113.78 AICc=33113.86 BIC=33157.98

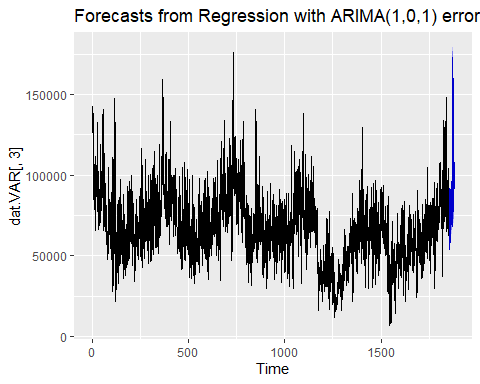
#incluir en lo anterior en internas todas las variables  
  
summary(fit)

## Series: dat.VAR[, 3]   
## Regression with ARIMA(1,0,1) errors   
##   
## Coefficients:  
## ar1 ma1 precio\_ref unidades\_total season pandemia  
## 0.9691 -0.5065 4.2676 965.4136 180.0001 -129.120  
## s.e. 0.0070 0.0231 1.7882 2.3587 195.1602 1031.595  
## precio  
## -2208.5236  
## s.e. 702.1504  
##   
## sigma^2 = 3325251: log likelihood = -16548.89  
## AIC=33113.78 AICc=33113.86 BIC=33157.98  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 7.979888 1820.081 1201.403 -0.05625762 1.917699 0.07857132  
## ACF1  
## Training set 0.01169076

#cargamos los datos de test que conocemos del proximo mes  
library(readr)  
varial <- read\_delim("varial.csv", delim = ";",   
 escape\_double = FALSE, trim\_ws = TRUE)

## Rows: 28 Columns: 7  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ";"  
## dbl (6): precio\_ref, unidades\_total, monto\_total, season, pandemia, precio  
## date (1): ds  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

varial = cbind(precio\_ref=varial$precio\_ref, unidades\_total=varial$unidades\_total,monto\_total=varial$monto\_total, season=varial$season,pandemia=varial$pandemia,precio=varial$precio)  
  
#predecimos los 28 dias de febrero  
 fcast <- forecast(fit, xreg=varial[,-3],h = 28)  
autoplot(fcast)

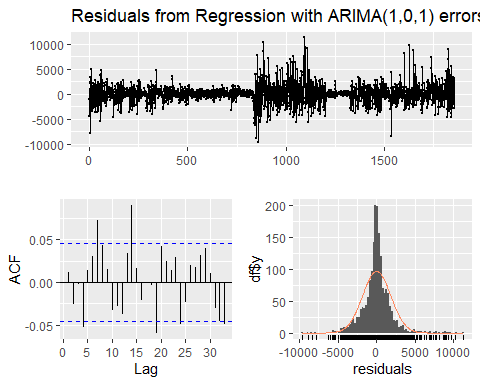


comparacion = cbind(fcast$mean,varial[,3])  
  
mean(abs((comparacion[,2]-comparacion[,1])/comparacion[,2])) \* 100

## [1] 3.302393

Obtenemos un error de 3.3, el modelo es bueno.

checkresiduals(fit)



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(1,0,1) errors  
## Q\* = 24.205, df = 3, p-value = 2.264e-05  
##   
## Model df: 7. Total lags used: 10

accuracy(fcast)

## ME RMSE MAE MPE MAPE MASE  
## Training set 7.979888 1820.081 1201.403 -0.05625762 1.917699 0.07857132  
## ACF1  
## Training set 0.01169076

Para el modelo actual, el valor MAPE es 1.91. Se indica que la diferencia absoluta promedio entre el valor pronosticado y el valor original es 1.91% aproximadamente.