

# RLM SERVICIOS DIARIOS

Miguel Angel Villegas

2025-03-11

```
library(tsDyn), library(tidyverse),library(dplyr), library(readxl), library(EnvStats), library(corrplot), library(caTools), library(GGally), library(forecast)
```

## Introducción

Este modelo abarca todas las ventas de los servicios, se utilizan las variables: Valor Unitario y Cantidad. Se realizan las correlaciones entre las tres variables y se determina el modelo lineal múltiple. El conjunto de entrenamiento y prueba esta dividido en una proporción de 80/20, sin embargo, la división es aleatoria, lo hace que el resultado sea mas confiable. Se siembra una semilla para permitir que los valores de la muestra sean los mismos.

```
ruta <- "/cloud/project/Ventas_Suministros_Totales.xlsx"
excel_sheets(ruta)
```

```
## [1] "Ventas Totales Original"      "Servicios Totales Original"
```

```
# "Ventas Totales Original"      "Servicios Totales Original"
```

```
Servicios_Totales <- as.data.frame(read_xlsx(ruta,
                                             sheet = "Servicios Totales Original"))
Servicios_Totales$Semana <- format(Servicios_Totales$Fecha, format = "%Y-%U")
Servicios_Totales$mes <- format(Servicios_Totales$Fecha, format = "%Y-%m")
```

```
Servicios_Totales <- Servicios_Totales %>%
  group_by(Fecha = as.Date(Fecha)) %>%
  summarize(Ventas_Totales = sum(Total),
            Ventas_Unitario = sum(ValorUnitario),
            Ventas_Cantidad = sum(Cantidad),
            .groups = "keep")
head(Servicios_Totales)
```

```
## # A tibble: 6 x 4
## # Groups:   Fecha [6]
##   Fecha      Ventas_Totales Ventas_Unitario Ventas_Cantidad
##   <date>          <dbl>          <dbl>          <dbl>
## 1 2019-07-01      20590          4550           27
## 2 2019-07-03      10962           450            21
## 3 2019-07-11       2494          2150             2
## 4 2019-07-20       8839          7620             1
## 5 2019-07-22      11484           450            22
## 6 2019-07-31     24457          8828             5
```

Seleccionando.

```
Servicios_Totales <- data.frame(Ventas_Totales = Servicios_Totales$Ventas_Totales, Ventas_Unitario = Se
head(Servicios_Totales)
```

```
## Ventas_Totales Ventas_Unitario Ventas_Cantidad
## 1      20590.00          4550          27
## 2      10962.00           450          21
## 3       2494.00          2150           2
## 4       8839.20          7620           1
## 5      11484.00           450          22
## 6      24457.44          8828           5
```

Se calcula el valor del parámetro lambda para la transformación Boxcox.

```
VS_lambda_dia <- boxcox(Servicios_Totales$Ventas_Totales,
                        objective.name = "Log-Likelihood", optimize = T)
# 0.05740459
VUs_lambda_dia <- boxcox(Servicios_Totales$Ventas_Unitario,
                        objective.name = "Log-Likelihood", optimize = T)
# 0.08905469
VCs_lambda_dia <- boxcox(Servicios_Totales$Ventas_Cantidad,
                        objective.name = "Log-Likelihood", optimize = T)
# -0.7605906
```

Se obtiene la transformación boxcox para los servicios, valor unitario de los servicios y cantidad de venta de los servicios, todos por día.

```
RLM_serv_dia <- Servicios_Totales %>%
  mutate(
    Ventas_Totales = boxcoxTransform(Servicios_Totales$Ventas_Totales, lambda = 0.05740459),
    Ventas_Unitario = boxcoxTransform(Servicios_Totales$Ventas_Unitario, lambda = 0.08905469),
    Ventas_Cantidad = boxcoxTransform(Servicios_Totales$Ventas_Cantidad, lambda = -0.7605906)
  )
head(RLM_serv_dia)
```

```
## Ventas_Totales Ventas_Unitario Ventas_Cantidad
## 1      13.388751      12.545165      1.2075746
## 2      12.293823       8.118417      1.1849956
## 3       9.872762      11.009803      0.5387200
## 4      11.928945      13.662355      0.0000000
## 5      12.373279       8.118417      1.1895070
## 6      13.694683      13.990694      0.9282059
```

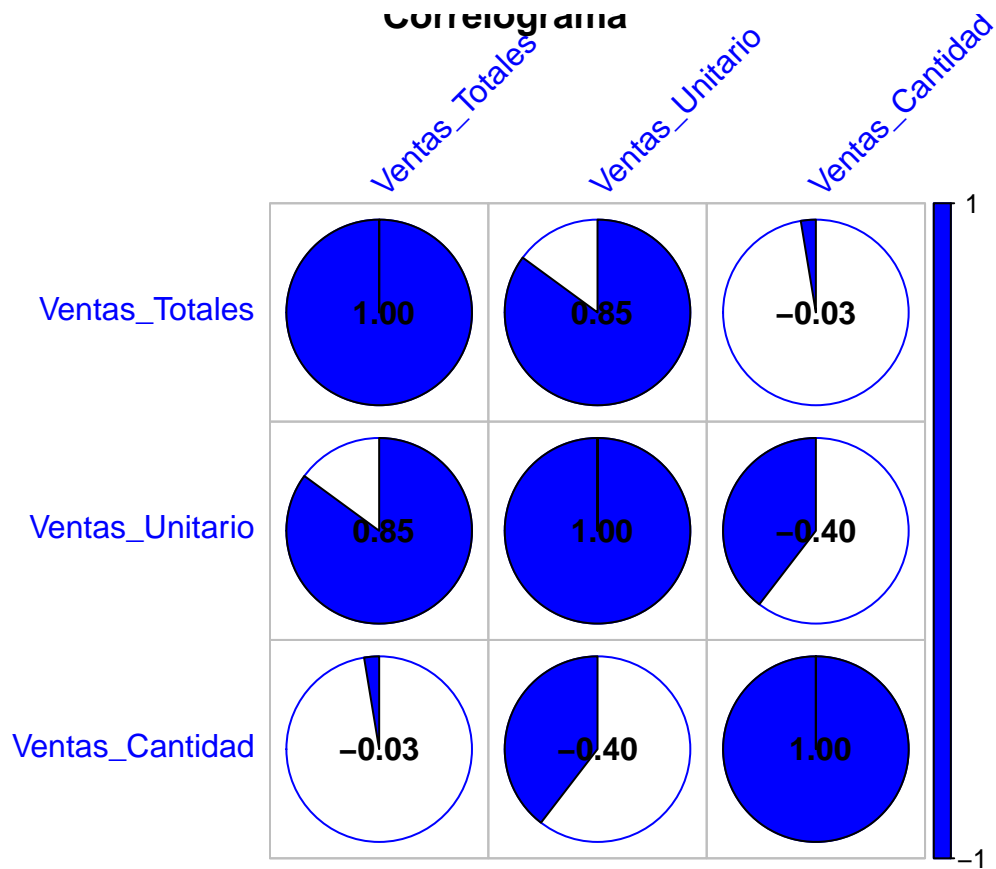
Se crea una matriz para el cálculo y visualización de las correlaciones, además se siembra la semilla para garantizar que los valores sean los mismos.

```
set.seed(123)
RLM_serv_dia_mtx <- cbind(RLM_serv_dia$Ventas_Totales,
                          RLM_serv_dia$Ventas_Unitario,
                          RLM_serv_dia$Ventas_Cantidad)
colnames(RLM_serv_dia_mtx) <- c("Ventas_Totales", "Ventas_Unitario", "Ventas_Cantidad")
head(RLM_serv_dia_mtx)
```

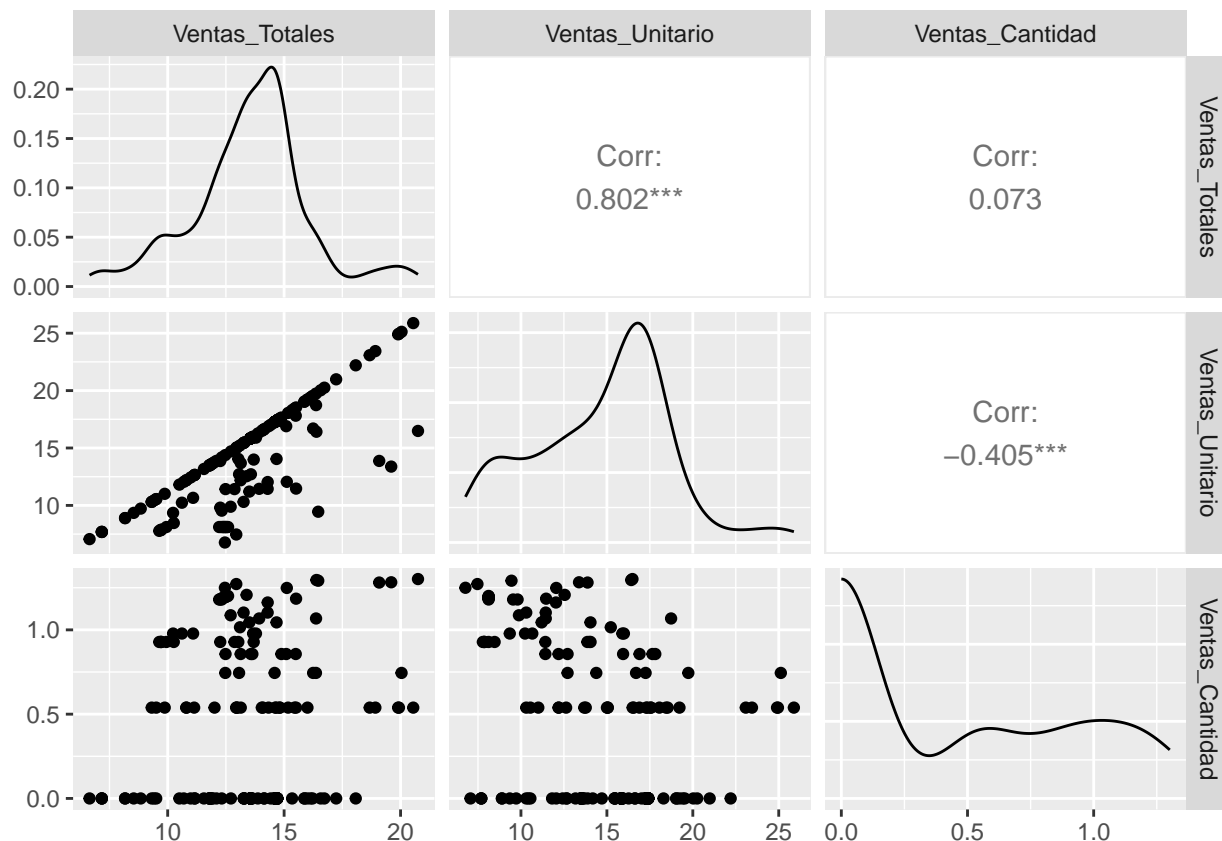
```
## Ventas_Totales Ventas_Unitario Ventas_Cantidad
## [1,]      13.388751      12.545165      1.2075746
## [2,]      12.293823       8.118417      1.1849956
## [3,]       9.872762      11.009803      0.5387200
## [4,]      11.928945      13.662355      0.0000000
```

```
## [5,]      12.373279      8.118417      1.1895070
## [6,]      13.694683     13.990694      0.9282059
```

## Gráficas de correlación



```
RLM_serv_dia <- as.data.frame(RLM_serv_dia_mtx)
RLM_serv_dia %>% GGally::ggpairs(cardinality_threshold = 10)
```



Se hace la división del conjunto de datos en una proporción de 80-20.

```
ms_lm <- sample.split(RLM_serv_dia$Ventas_Totales, SplitRatio = 0.80)
es_lm <- subset(RLM_serv_dia, ms_lm == T)
ps_lm <- subset(RLM_serv_dia, ms_lm == F)
```

## Modelo

Se determina el modelo lineal.

```
Moslm <- lm(Ventas_Totales ~ ., data = es_lm)
```

Resumen del modelo obtenido.

```
print(summary(Moslm))
```

```
##
## Call:
## lm(formula = Ventas_Totales ~ ., data = es_lm)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0734 -0.7261  0.2291  0.4174  4.7764
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.26531    0.36482    8.95 1.62e-15 ***
## Ventas_Unitario 0.63310    0.02162   29.29 < 2e-16 ***
## Ventas_Cantidad 2.40697    0.18347   13.12 < 2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9642 on 144 degrees of freedom
## Multiple R-squared:  0.8565, Adjusted R-squared:  0.8545
## F-statistic: 429.8 on 2 and 144 DF,  p-value: < 2.2e-16
# Multiple R-squared:  0.8565, Adjusted R-squared:  0.8545
```

## Pronostico

```
pronostico_lms <- predict(Moslm, ps_lm)
```

Se crea una data frame con los resultados y los valores actuales.

```
resultados_s <- cbind(pronostico_lms, ps_lm$Ventas_Totales)
resultados_s <- as.data.frame(resultados_s)
colnames(resultados_s) <- c("prediccion", "actual")
head(resultados_s)
```

```
##      prediccion      actual
## 2      11.25735 12.293823
## 4      11.91498 11.928945
## 5      11.26821 12.373279
## 8      12.55675 11.136800
## 11     11.24093  9.507298
## 16     11.28752 12.522466
```

Si es hay valores menores que cero se substituyen por cero.

```
any(resultados_s < 0)
```

```
## [1] FALSE
```

Función.

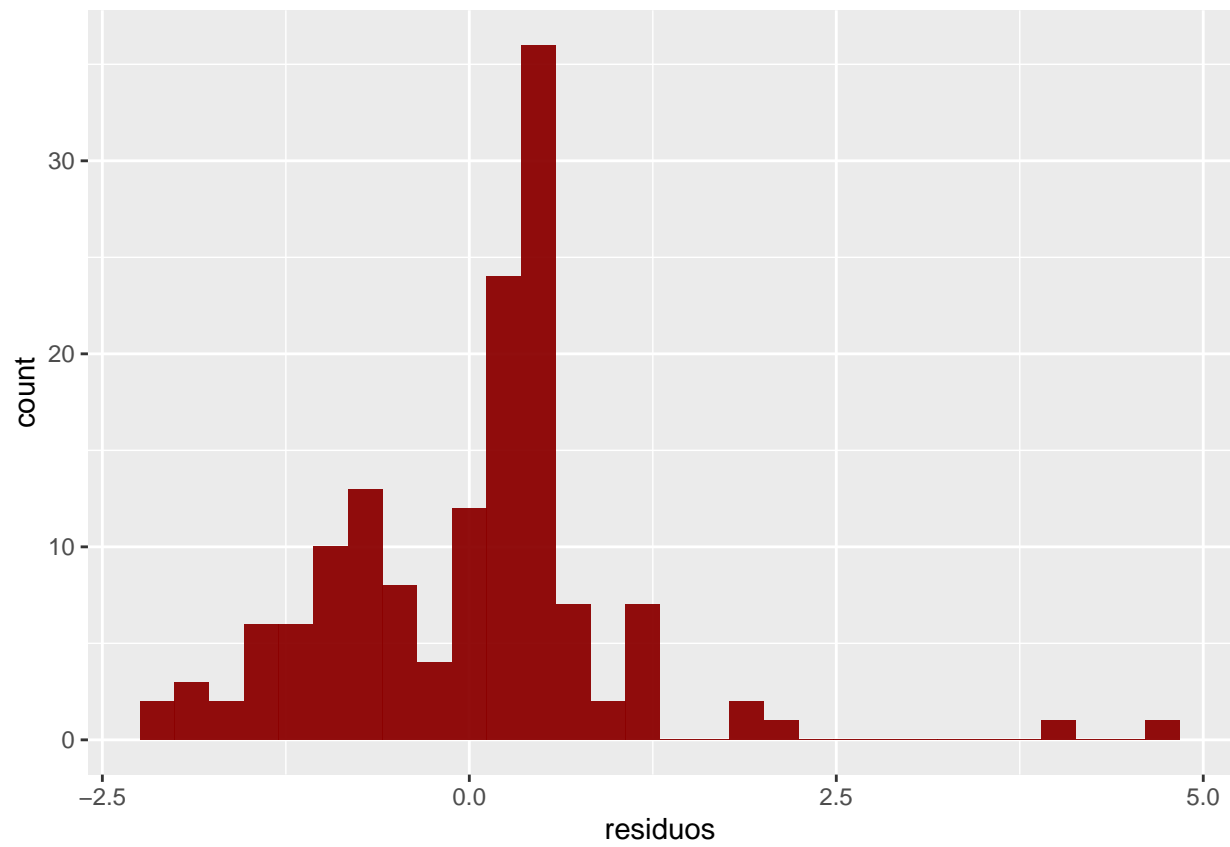
Exactitud del modelo.

```
summary(Moslm)$r.squared
```

```
## [1] 0.8565253
```

```
# [1] 0.8565253
```

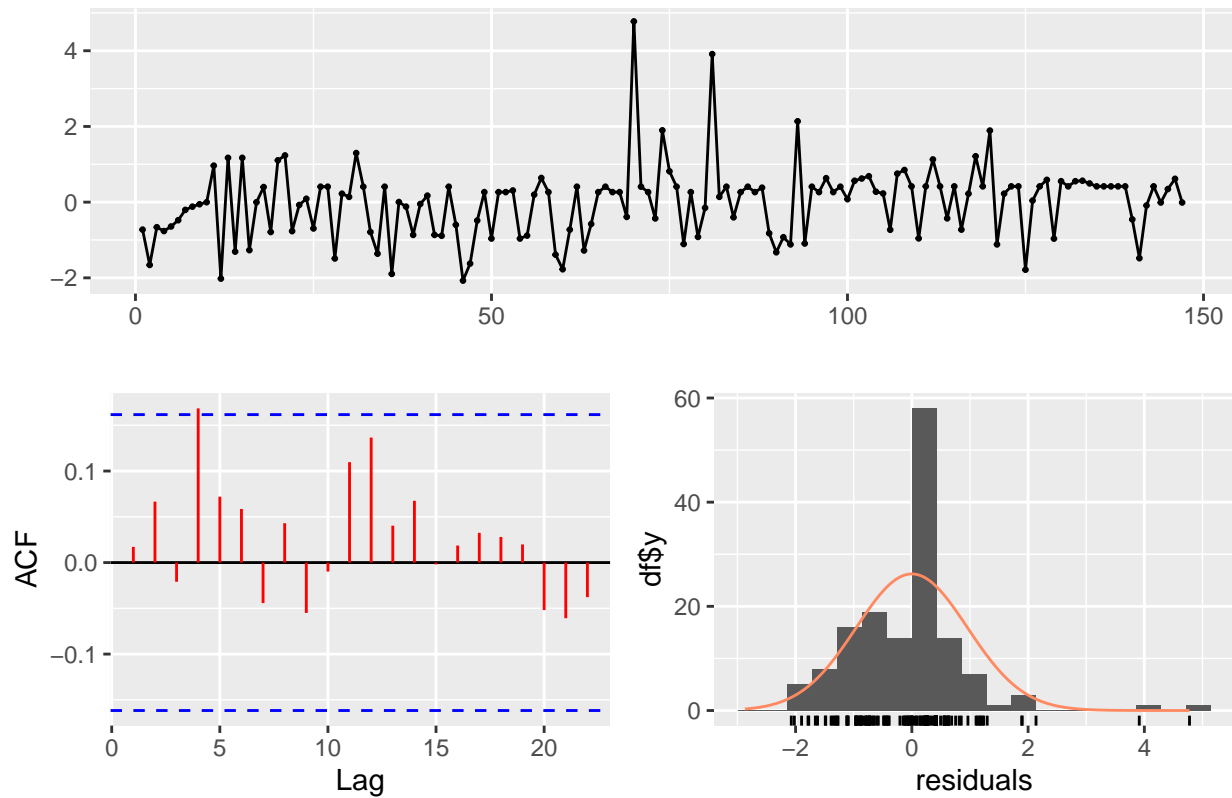
Inspección de los residuales.



Residuales

```
checkresiduals(Moslm, col = "red")
```

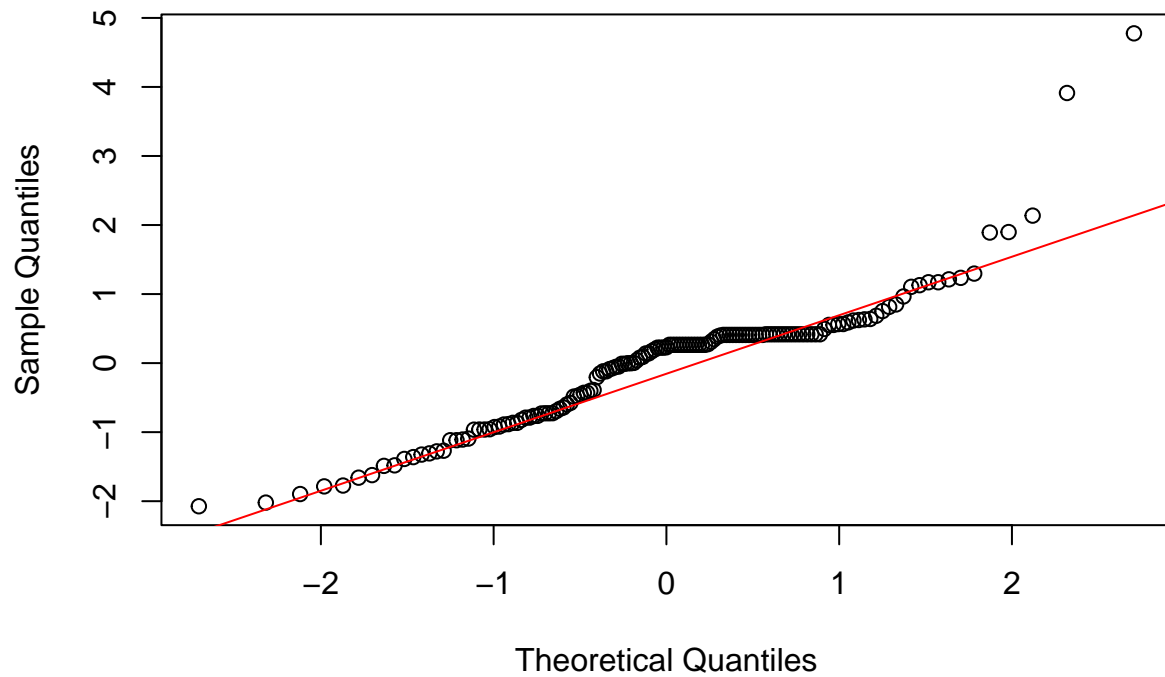
## Residuals



```
##  
## Breusch-Godfrey test for serial correlation of order up to 10  
##  
## data: Residuals  
## LM test = 7.1421, df = 10, p-value = 0.712  
# p-value = 0.712
```

Inspeccionando si existe normalidad en los residuales

## Normal Q-Q Plot

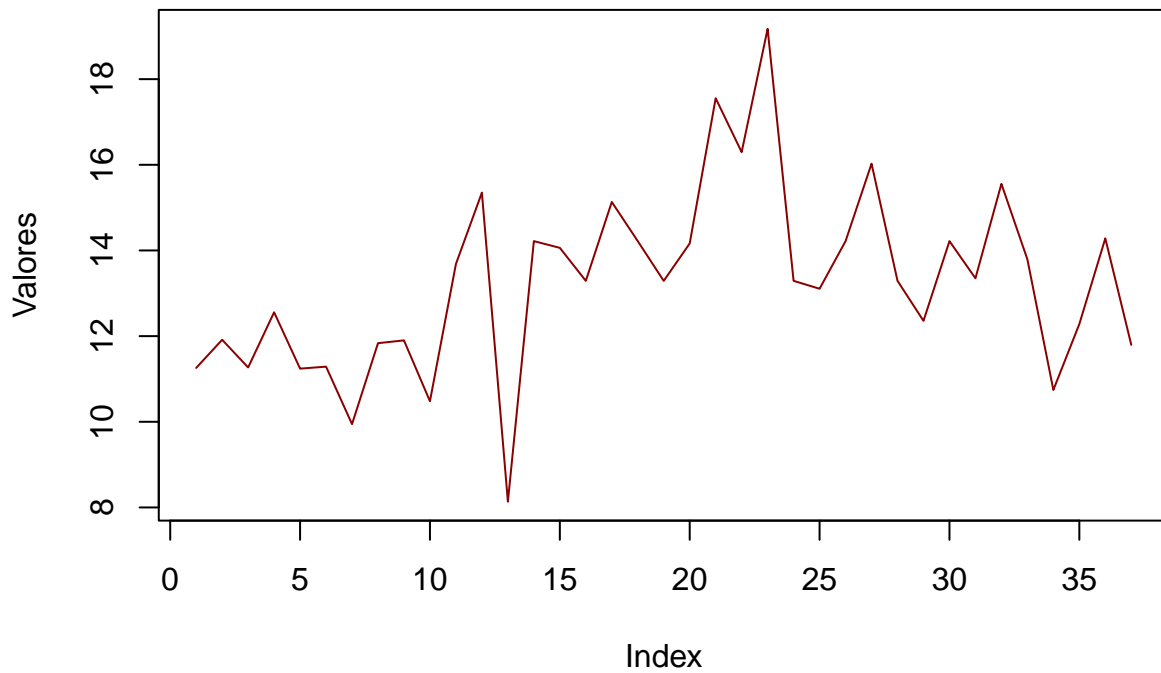


Se grafican los pronósticos.

```
pronostico_serv_lm <- predict(Moslm, ps_lm)
```

```
plot(pronostico_serv_lm, type = "lines", col = "darkred", ylab = "Valores")
```

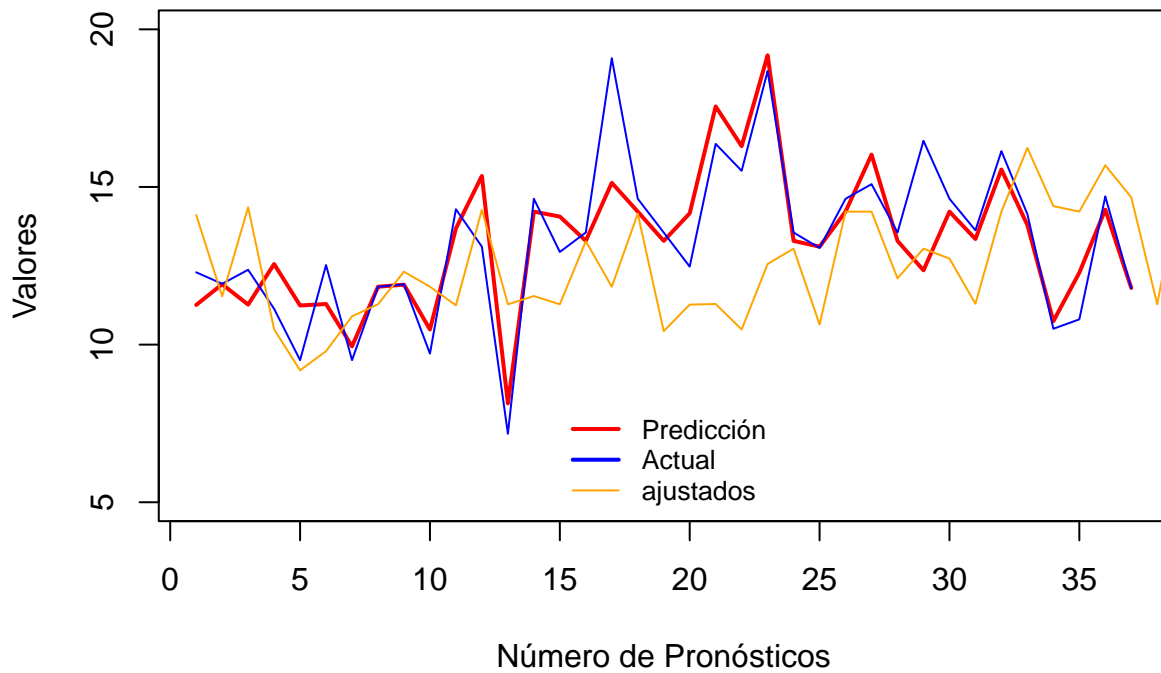
```
## Warning in plot.xy(xy, type, ...): plot type 'lines' will be truncated to first  
## character
```



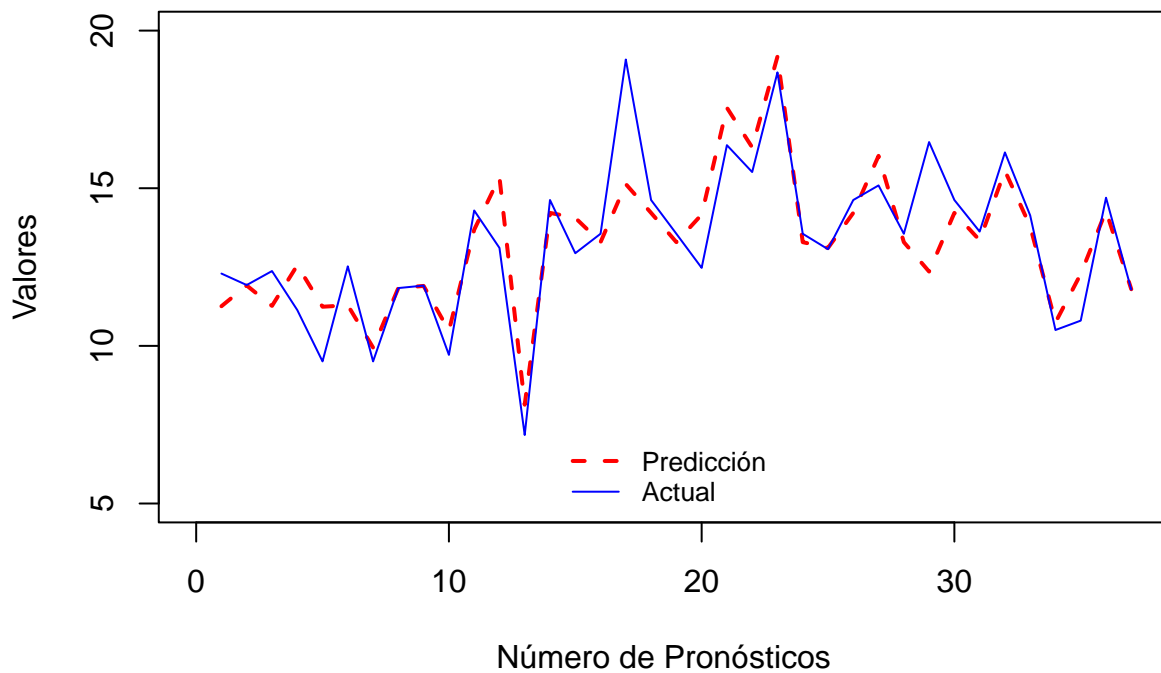


Gráfica de los pronósticos junto con los valores reales.

### Predicción vs Actual por Regresión Multi-Lineal



### Predicción vs Actual por Regresión Multi-Lineal



Se determina la exactitud del modelo.

```
accuracy(resultados_s$prediccion[1:37], ps_lm$Ventas_Totales[1:37])
```

##		ME	RMSE	MAE	MPE	MAPE
##	Test set	0.0223222	1.269369	0.8632803	-0.7184874	6.515798

#		<i>ME</i>	<i>RMSE</i>	<i>MAE</i>	<i>MPE</i>	<i>MAPE</i>
#	Test set	0.0223222	1.269369	0.8632803	-0.7184874	6.515798

## Conclusiones

El modelo captura 86.97% la dinámica de la serie, los residuales están muy por encima del valor ideal de  $p > 0.05$ , lo que indica que no existe una correlación entre los residuos.