TALLER MODELO SEMMA Y TDSP

Daniel Arteta Salazar

2025-11-10

Librerías

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.3.0
## v purrr     1.1.0     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

GENERACIÓN DE DATOS: RIESGO DE SINIESTRO (ACTUARIAL)

```
set.seed(123)   # para reproducibilidad
n <- 3000        # tamaño de la muestra

# Variables principales
edad_conductor <- sample(18:75, n, replace = TRUE)
antiguedad_vehiculo <- sample(0:20, n, replace = TRUE)
valor_vehiculo <- round(runif(n, 20, 120), 1)  # millones
region <- sample(c("Urbana", "Rural"), n, replace = TRUE, prob = c(0.7, 0.3))

# Generación de siniestros
# modelo base para probabilidad de siniestro (logit)
# conductores más jóvenes y autos más viejos = mayor riesgo
logit_p <- -4 + 0.05*(60 - edad_conductor) +
  0.08*antiguedad_vehiculo +
  0.6*(region == "Urbana")

# convertir logit a probabilidad (logística inversa)
p_siniestro <- exp(logit_p) / (1 + exp(logit_p))

# número de siniestros según probabilidad esperada
num_siniestros <- rpois(n, lambda = p_siniestro * 2)

# variable binaria de siniestro (1 si hay al menos uno)
siniestro <- ifelse(num_siniestros > 0, 1, 0)
```

```r
# Base de datos final
datos_siniestros <- data.frame(
  edad_conductor,
  antiguedad_vehiculo,
  valor_vehiculo,
  region,
  num_siniestros,
  siniestro
)

# Vista previa
head(datos_siniestros)
```

```
##   edad_conductor antiguedad_vehiculo valor_vehiculo region num_siniestros
## 1             48                   5           45.1 Urbana              1
## 2             32                   9           47.8 Urbana              0
## 3             68                   9           89.1 Urbana              0
## 4             31                   6           27.6  Rural              0
## 5             20                  10           36.1  Rural              2
## 6             59                  16           70.1  Rural              0
##   siniestro
## 1         1
## 2         0
## 3         0
## 4         0
## 5         1
## 6         0
```

PARTE A - MODELO SEMMA

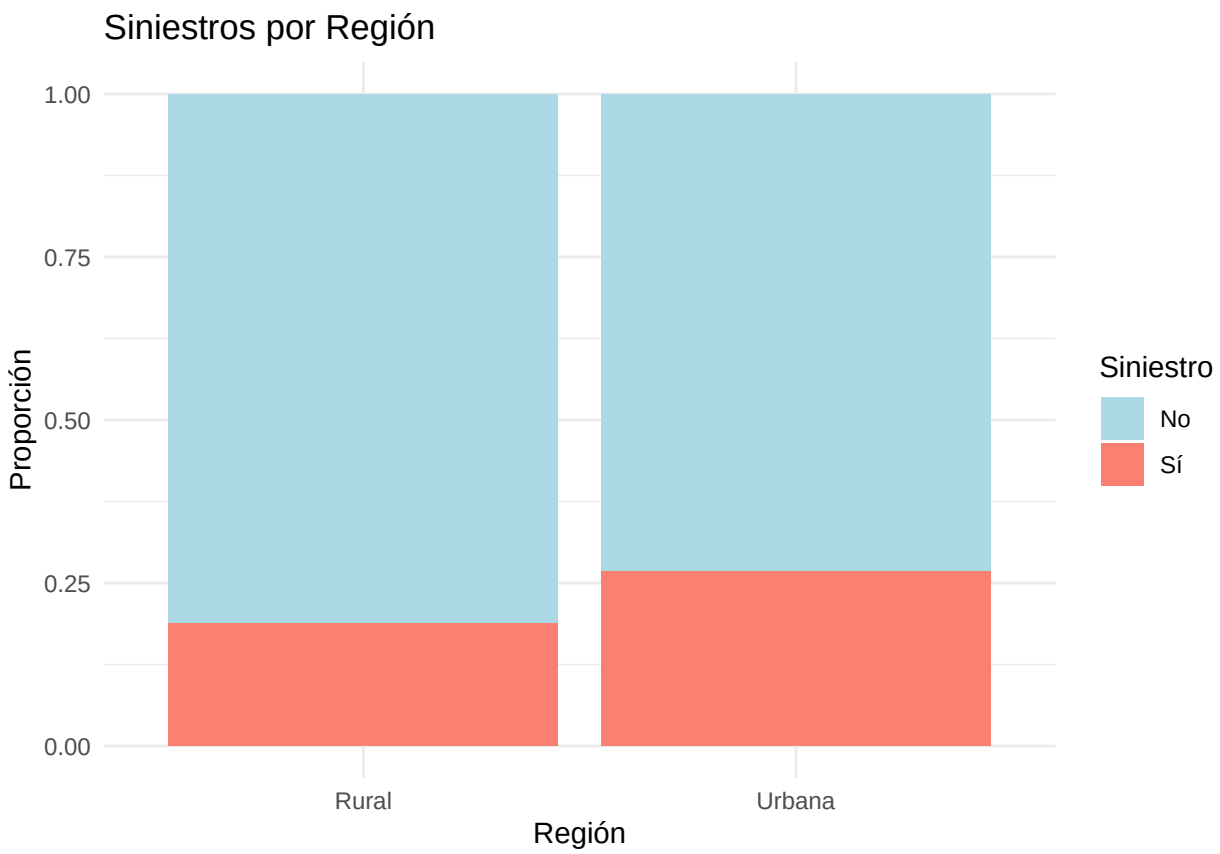
A.1 Sample

```r
set.seed(123)
n <- nrow(datos_siniestros)
indice <- sample(1:n, size = 0.7 * n)
train <- datos_siniestros[indice, ]
test <- datos_siniestros[-indice, ]
```

La separación de datos evita el sobreajuste, permitiendo evaluar el modelo en datos no vistos. Esto asegura que el modelo sea generalizable y no solo memorice la muestra de entrenamiento.

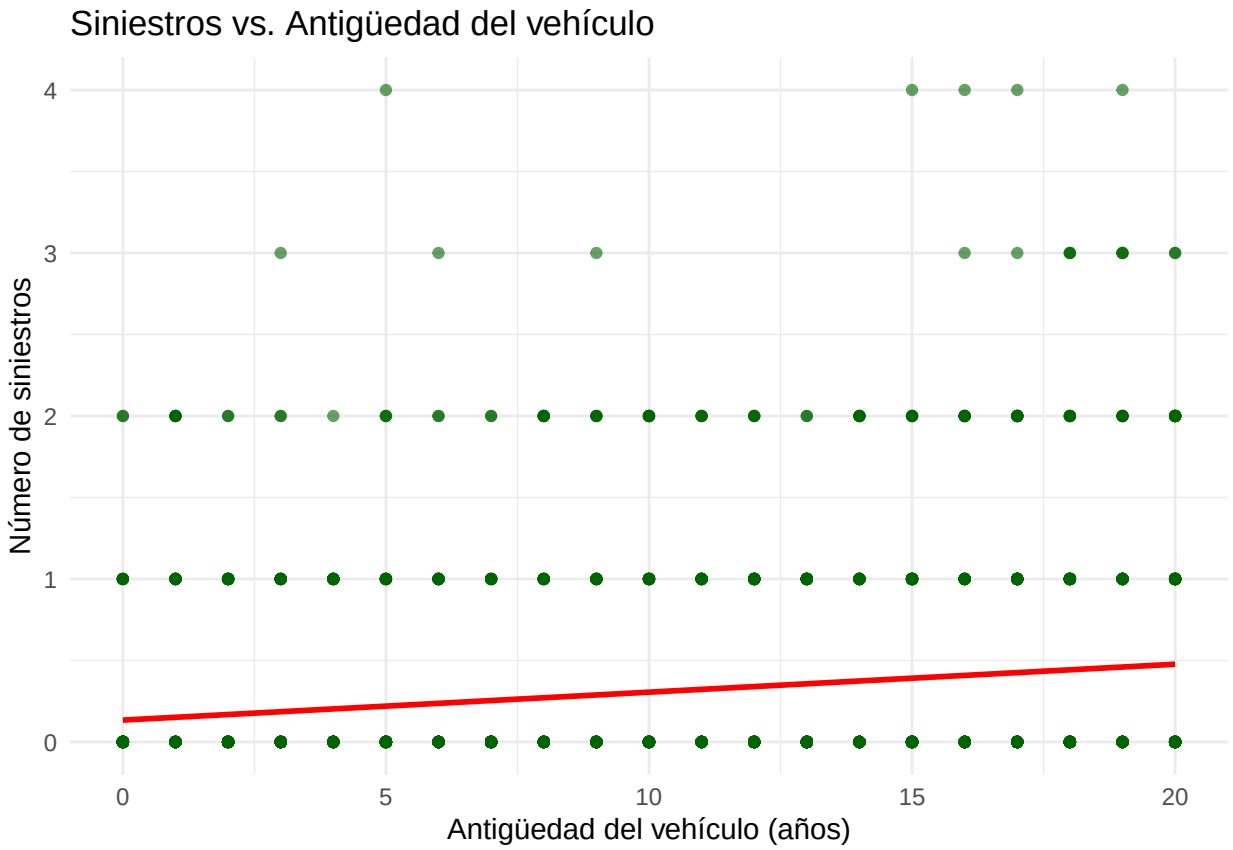
A.2 Explore

```r
# Siniestros por región
ggplot(train, aes(x = region, fill = factor(siniestro))) +
  geom_bar(position = "fill") +
  labs(title = "Siniestros por Región",
       x = "Región",
       y = "Proporción",
       fill = "Siniestro") +
  scale_fill_manual(values = c("0" = "lightblue", "1" = "salmon"),
                    labels = c("No", "Sí")) +
  theme_minimal()
```

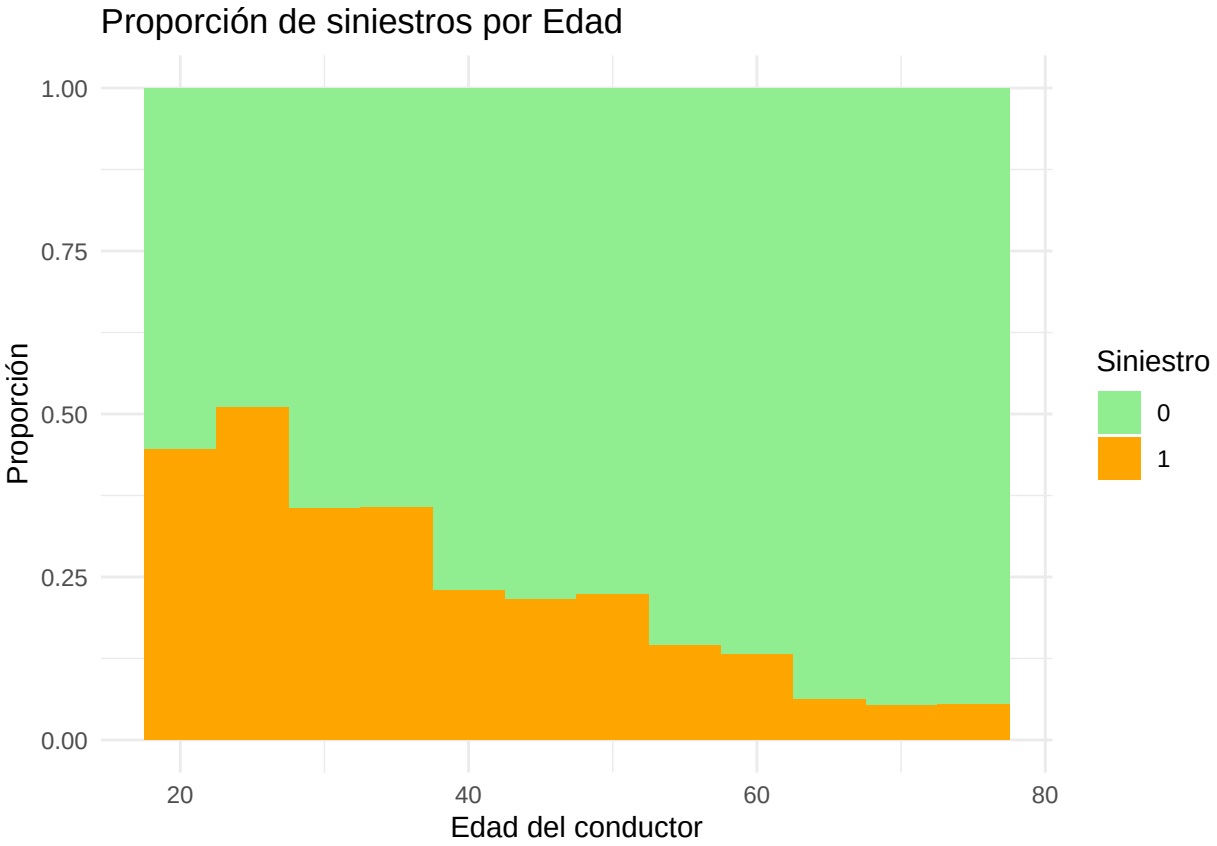



```
# Siniestros por antigüedad del vehículo
ggplot(train, aes(x = antiguedad_vehiculo, y = num_siniestros)) +
  geom_point(alpha = 0.6, color = "darkgreen") +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(title = "Siniestros vs. Antigüedad del vehículo",
       x = "Antigüedad del vehículo (años)",
       y = "Número de siniestros") +
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# Distribución de siniestros por grupo de edad
ggplot(train, aes(x = edad_conductor, fill = factor(siniestro))) +
  geom_histogram(binwidth = 5, position = "fill") +
  labs(title = "Proporción de siniestros por Edad",
       x = "Edad del conductor",
       y = "Proporción",
       fill = "Siniestro") +
  scale_fill_manual(values = c("0" = "lightgreen", "1" = "orange")) +
  theme_minimal()
```



Observaciones:

- Los conductores más jóvenes (18-25 años) presentan mayor frecuencia de siniestros.
- La región urbana muestra mayor proporción de siniestros comparado con la región rural.
- Los vehículos más antiguos (>10 años) tienen mayor número de siniestros.
- Existe una relación no lineal entre edad y siniestralidad.

A.3 Modify

```
# Grupo de edad (joven: <25 años, adulto: 25-65, mayor: >65)
train <- train %>%
  mutate(grupo_edad = case_when(
    edad_conductor <= 25 ~ "Joven",
    edad_conductor <= 65 ~ "Adulto",
    TRUE ~  "Adulto mayor"
  )
)

test <- test %>%
  mutate(grupo_edad = case_when(
    edad_conductor <= 25 ~ "Joven",
    edad_conductor <= 65 ~ "Adulto",
    TRUE ~  "Adulto mayor"
  )
)

# Logaritmo del valor del vehículo (para reducir asimetría)
train$log_valor <- log(train$valor_vehiculo)
```

```r
test$log_valor <- log(test$valor_vehiculo)
```

Justificación:

- 'grupo_edad" captura mejor el riesgo asociado a la experiencia del conductor.
- `log_valor` linealiza la relación con la variable respuesta y reduce el efecto de valores extremos.

A.4 Model

```r
modelo_siniestros <- lm(num_siniestros ~ edad_conductor + antiguedad_vehiculo +
                        region + log_valor, data = train)
summary(modelo_siniestros)
```

```
##
## Call:
## lm(formula = num_siniestros ~ edad_conductor + antiguedad_vehiculo +
##     region + log_valor, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8674 -0.3713 -0.1383  0.1900  3.4852
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.7353975  0.1181263   6.226 5.78e-10 ***
## edad_conductor      -0.0130396  0.0007471 -17.454  < 2e-16 ***
## antiguedad_vehiculo  0.0186496  0.0020060   9.297  < 2e-16 ***
## regionUrbana         0.0937384  0.0267009   3.511 0.000456 ***
## log_valor           -0.0200813  0.0261712  -0.767 0.442987
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5608 on 2095 degrees of freedom
## Multiple R-squared:  0.1582, Adjusted R-squared:  0.1566
## F-statistic: 98.45 on 4 and 2095 DF,  p-value: < 2.2e-16
```

Signos esperados:

- `edad_conductor`: coeficiente positivo (mayor siniestros con mayor edad).
- `antiguedad_vehiculo`: coeficiente positivo (más siniestros en autos viejos).
- `region`: coeficiente positivo en la región urbana (mayor riesgo en zona urbana).

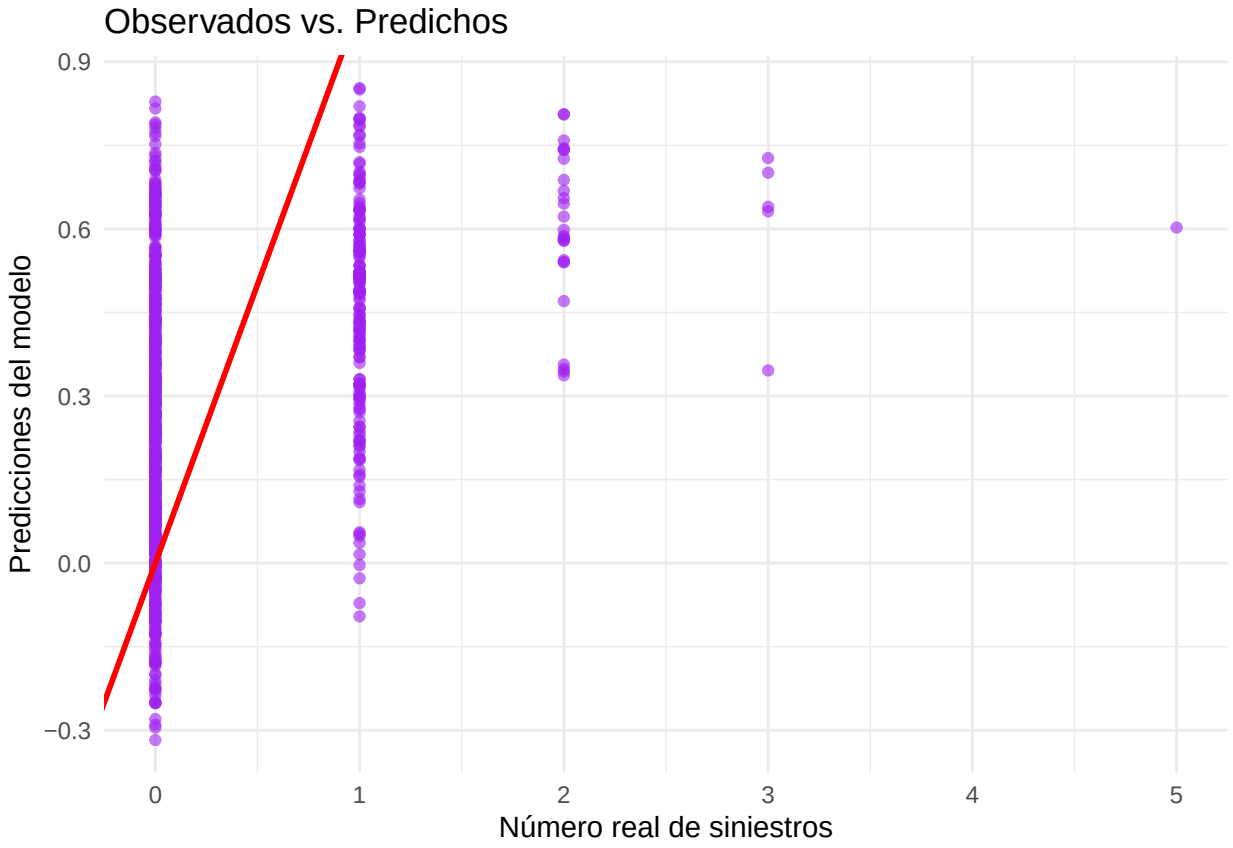
A.5 Asses

```r
predicciones <- predict(modelo_siniestros, newdata = test)
r2 <- 1 - sum((test$num_siniestros - predicciones)^2) /
           sum((test$num_siniestros - mean(test$num_siniestros))^2)
r2
```

```
## [1] 0.1375666
```

```r
# Gráfico de observados vs. predichos
ggplot(data = test, aes(x = num_siniestros, y = predicciones)) +
  geom_point(alpha = 0.6, color = "purple") +
  geom_abline(intercept = 0, slope = 1, color = "red", linewidth = 1) +
  labs(title = "Observados vs. Predichos",
```

```
      x = "Número real de siniestros",
      y = "Predicciones del modelo") +
  theme_minimal()
```



Interpretación:

El R^2 indica que el modelo solo explica una porción pequeña de la variabilidad del número de siniestros.

PARTE B - MODELO TDSP

B.1 Bussiness understanding

Objetivo: Predecir si ocurrirá al menos un siniestro en el año de la póliza.

Métrica de éxito:

- Exactitud mayor o igual a 70%
- Sensibilidad mayor o igual a 60%

Umbral inicial: 0.5

Implicaciones de clasificación errónea:

- Falso negativo: No identificar un cliente de alto riesgo => pérdidas económicas.
- Falso positivo: Sobreestimar el riesgo => posible pérdida de clientes por primas elevadas.

B.2 Data understanding

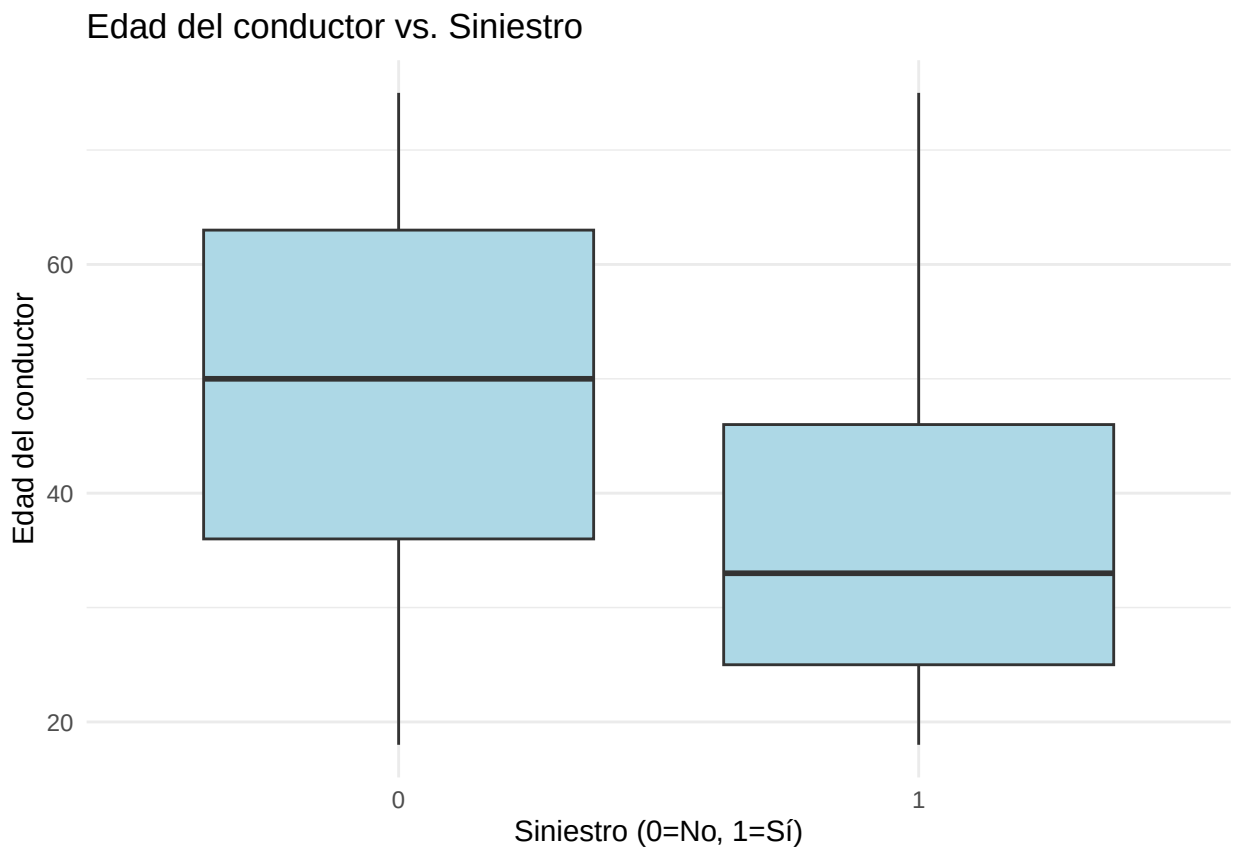
```
# Resumen estadístico
summary(datos_siniestros)
```

```
##  edad_conductor  antiguedad_vehiculo valor_vehiculo      region
##  Min.   :18.00   Min.   : 0.00       Min.   : 20.00   Length:3000
##  1st Qu.:32.00   1st Qu.: 5.00       1st Qu.: 45.70   Class :character
##  Median :46.00   Median :10.00       Median : 69.35   Mode  :character
##  Mean   :46.09   Mean   :10.09       Mean   : 69.70
##  3rd Qu.:60.00   3rd Qu.:15.00       3rd Qu.: 93.83
##  Max.   :75.00   Max.   :20.00       Max.   :120.00
##  num_siniestros     siniestro
##  Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.000
##  Median :0.0000   Median :0.000
##  Mean   :0.2947   Mean   :0.235
##  3rd Qu.:0.0000   3rd Qu.:0.000
##  Max.   :5.0000   Max.   :1.000
```

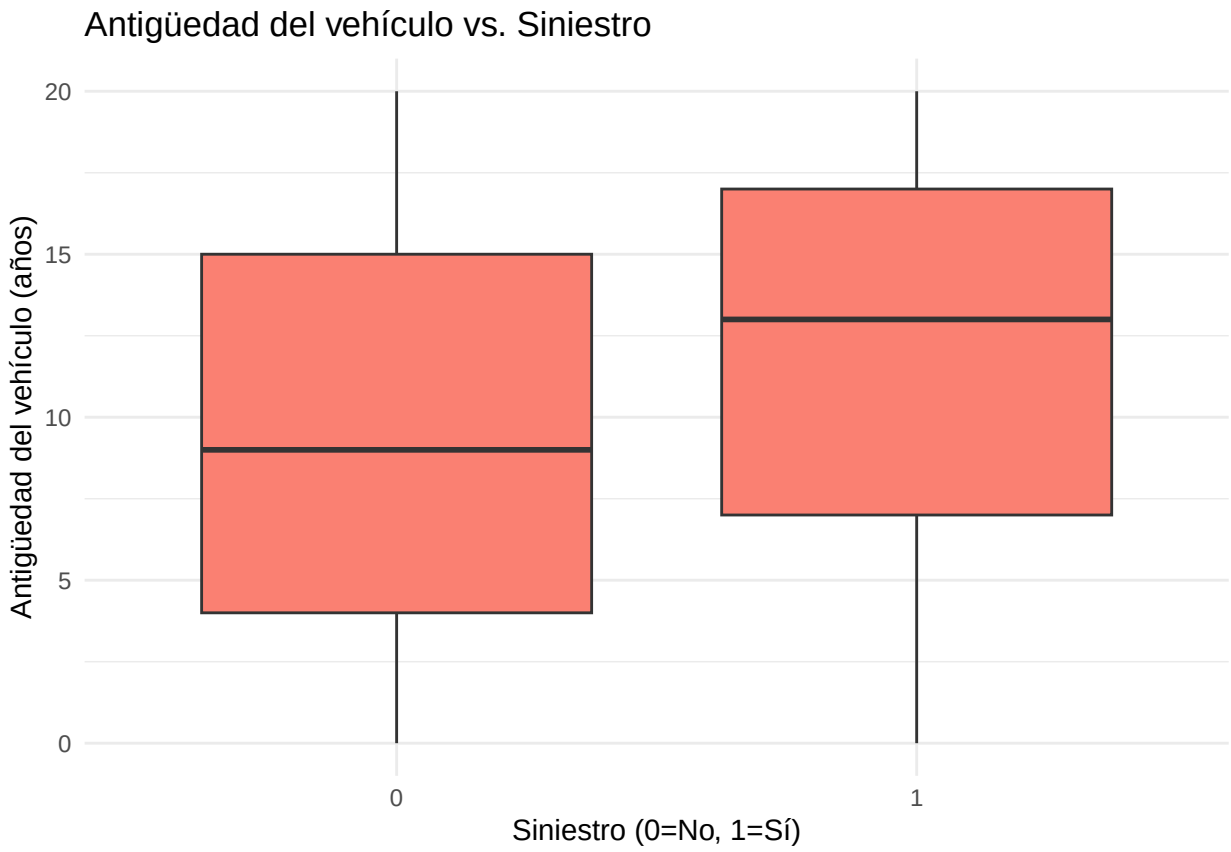
```r
# Proporción de siniestros
mean(datos_siniestros$siniestro)
```

```
## [1] 0.235
```

```r
# Gráfico de distribución de variables principales
ggplot(datos_siniestros, aes(x = factor(siniestro), y = edad_conductor)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Edad del conductor vs. Siniestro",
       x = "Siniestro (0=No, 1=Sí)",
       y = "Edad del conductor") +
  theme_minimal()
```



```r
ggplot(datos_siniestros, aes(x = factor(siniestro), y = antiguedad_vehiculo)) +
  geom_boxplot(fill = "salmon") +
  labs(title = "Antigüedad del vehículo vs. Siniestro",
       x = "Siniestro (0=No, 1=Sí)",
       y = "Antigüedad del vehículo (años)") +
  theme_minimal()
```



Verificación:

- No hay valores faltantes o negativos.
- Todas las variables son conocidas antes del siniestro.
- Proporción de siniestros: ~30% (equilibrio aceptable).

B.3 Modeling

```r
modelo_logistico <- glm(siniestro ~ edad_conductor + antiguedad_vehiculo +
                        region + valor_vehiculo + grupo_edad,
                        data = train, family = binomial())
summary(modelo_logistico)
```

```
##
## Call:
## glm(formula = siniestro ~ edad_conductor + antiguedad_vehiculo +
##     region + valor_vehiculo + grupo_edad, family = binomial(),
##     data = train)
##
## Coefficients:
```

```
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              0.272785   0.312291   0.873 0.382393
## edad_conductor          -0.058021   0.005948  -9.755  < 2e-16 ***
## antiguedad_vehiculo      0.081387   0.009494   8.573  < 2e-16 ***
## regionUrbana             0.484484   0.127827   3.790 0.000151 ***
## valor_vehiculo          -0.002347   0.001964  -1.195 0.232186
## grupo_edadAdulto mayor -0.092360   0.289550  -0.319 0.749742
## grupo_edadJoven        -0.075583   0.182244  -0.415 0.678334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2330.5  on 2099  degrees of freedom
## Residual deviance: 1984.8  on 2093  degrees of freedom
## AIC: 1998.8
##
## Number of Fisher Scoring iterations: 5
```

Interpretación de coeficientes:

- `edad_conductor`: Signo negativo => menor probabilidad de siniestro con mayor edad.
- `antiguedad_vehiculo`: Signo positivo => mayor probabilidad con vehículos más viejos.
- `region`: Signo positivo => mayor riesgo en zona urbana.
- `grupo_edad`: Signo positivo => mayor riesgo en conductores jóvenes.

B.4 Assess

```r
# Función para calcular métricas
calcular_metricas <- function(probabilidades, real, umbral) {
  pred_clase <- ifelse(probabilidades >= umbral, 1, 0)
  exactitud <- mean(pred_clase == real)
  TP <- sum(pred_clase == 1 & real == 1)
  FN <- sum(pred_clase == 0 & real == 1)
  sensibilidad <- TP / (TP + FN)
  return(c(Exactitud = exactitud, Sensibilidad = sensibilidad))
}

prob_test <- predict(modelo_logistico, newdata = test, type = "response")

# Umbral 0.5
metricas_05 <- calcular_metricas(prob_test, test$siniestro, 0.5)
cat("Umbral 0.5 - Exactitud:", round(metricas_05[1], 3),
    "Sensibilidad:", round(metricas_05[2], 3))
```

```
## Umbral 0.5 - Exactitud: 0.773 Sensibilidad: 0.222
```

```r
# Umbral 0.3
metricas_03 <- calcular_metricas(prob_test, test$siniestro, 0.3)
cat("Umbral 0.3 - Exactitud:", round(metricas_03[1], 3),
    "Sensibilidad:", round(metricas_03[2], 3))
```

```
## Umbral 0.3 - Exactitud: 0.729 Sensibilidad: 0.624
```

```r
# Umbral 0.7
metricas_07 <- calcular_metricas(prob_test, test$siniestro, 0.7)
```

```r
cat("Umbral 0.7 - Exactitud:", round(metricas_07[1], 3),
    "Sensibilidad:", round(metricas_07[2], 3))
```

```
## Umbral 0.7 - Exactitud: 0.786 Sensibilidad: 0.01
```

```r
# Curva ROC y gráfico de probabilidades
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

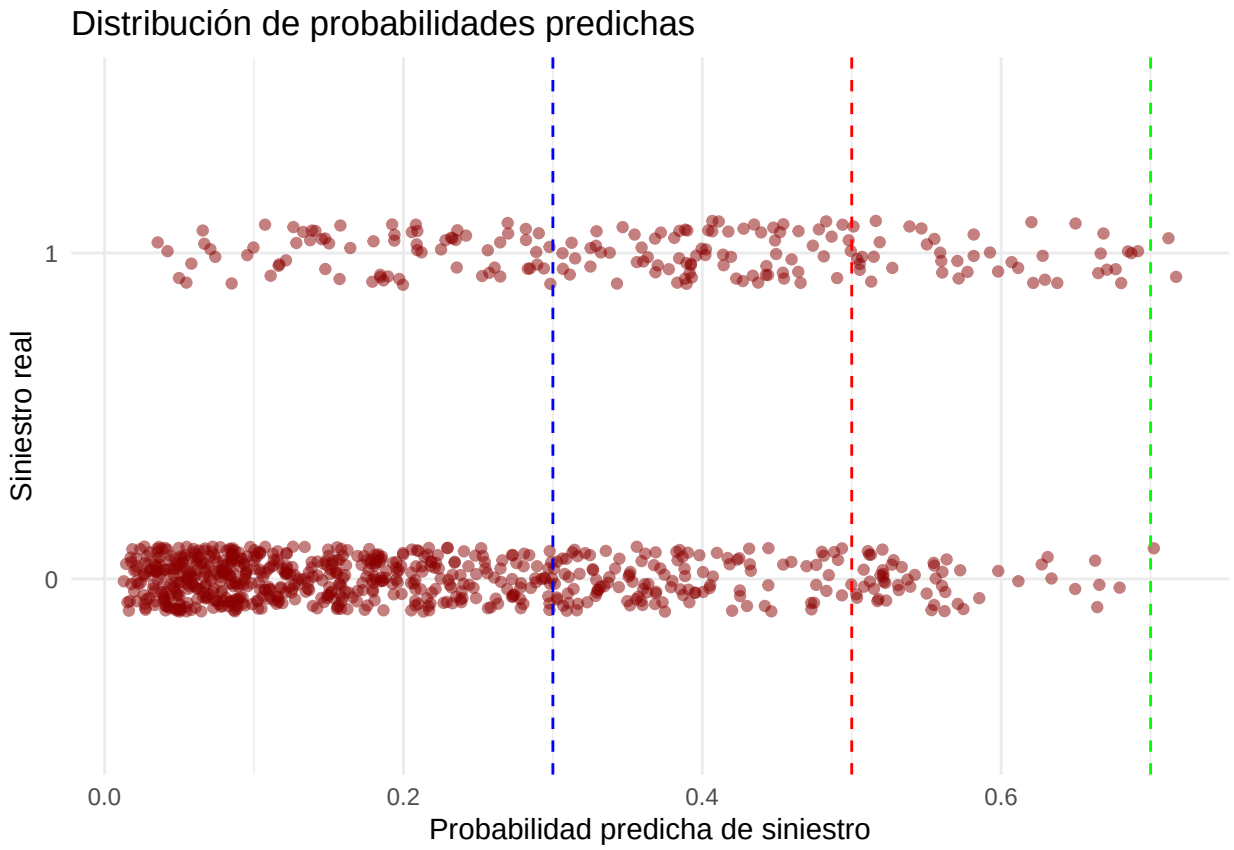
```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
roc_obj <- roc(test$siniestro, prob_test)
```

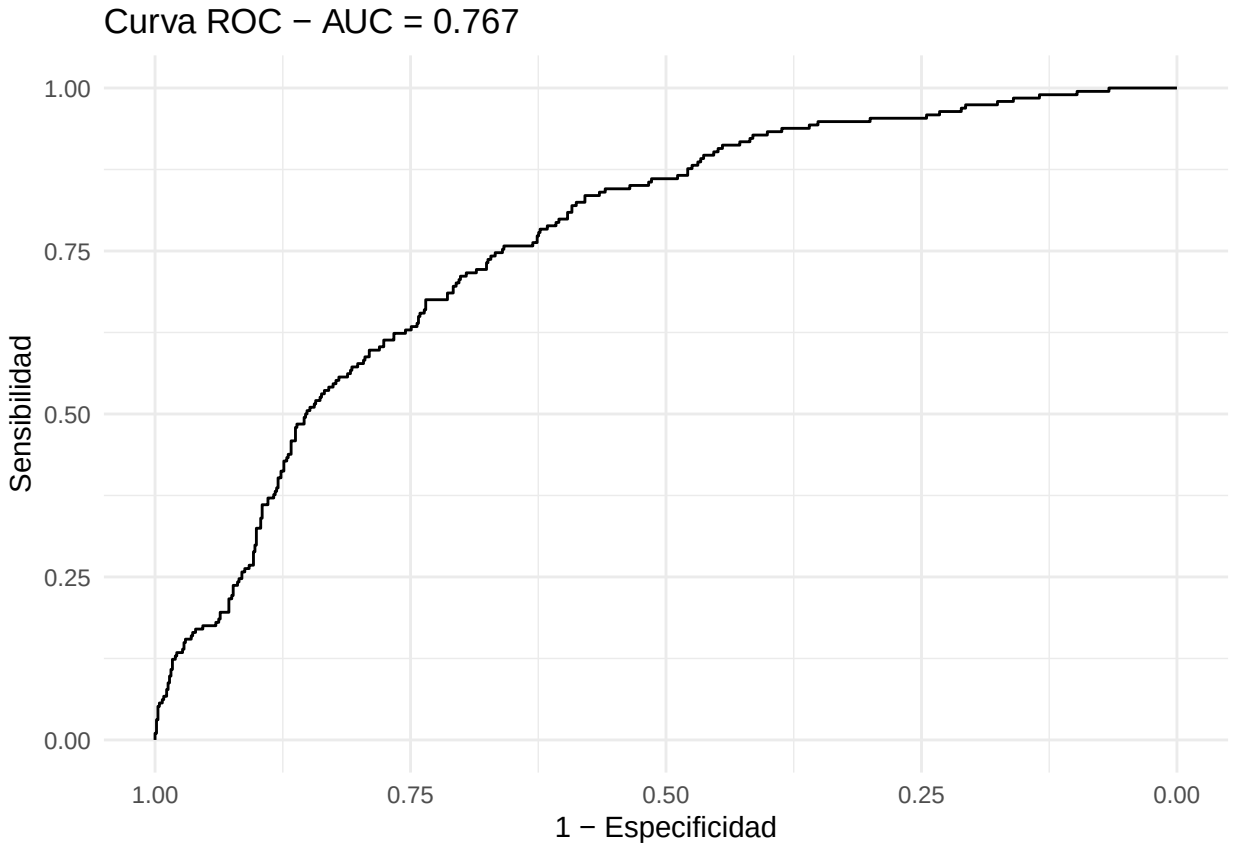
```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```r
auc_value <- auc(roc_obj)

p9 <- ggplot(data = test, aes(x = prob_test, y = factor(siniestro))) +
  geom_jitter(height = 0.1, alpha = 0.5, color = "darkred") +
  geom_vline(xintercept = c(0.3, 0.5, 0.7), linetype = "dashed",
             color = c("blue", "red", "green")) +
  labs(title = "Distribución de probabilidades predichas",
       x = "Probabilidad predicha de siniestro",
       y = "Siniestro real") +
  theme_minimal()

p10 <- ggroc(roc_obj) +
  labs(title = paste("Curva ROC - AUC =", round(auc_value, 3)),
       x = "1 - Especificidad",
       y = "Sensibilidad") +
  theme_minimal()

print(p9)
```


Distribución de probabilidades predichas
Siniestro real
1
0
0.0
0.2
0.4
0.6
Probabilidad predicha de siniestro

```
print(p10)
```



Para la aseguradora es más crítico evitar falsos negativos (no detectar alto riesgo), por lo que una mayor sensibilidad es preferible, incluso a costa de más falsos positivos. Un umbral de 0.3 proporciona mejor sensibilidad.

B.5 Deployment

Aplicaciones del modelo:

- Tarificación diferenciada: Ajustar primas según el riesgo predicho.
- Descuentos: Ofrecer reducciones a clientes de bajo riesgo.
- Prevención: Identificar clientes de alto riesgo para programas de educación vial.

Limitaciones:

- Muestra limitada a 3000 registros.
- Posibles variables omitidas (ej: historial de conducción, kilometraje anual).
- Riesgo de sobregeneralización si no se actualiza el modelo periódicamente.