Universidad Nacional Autónoma de México

Facultad de Ciencias

Algoritmos de aprendizaje automático aplicados a la clasificación binaria

T E S I S

QUE PARA OBTENER POR EL GRADO DE:

LICENCIADO EN MATEMÁTICAS APLICADAS

PRESENTA:

RAÚL LÓPEZ BENÍTEZ

DIRECTOR DE TESIS:
DRA. RUTH SELENE FUENTES GARCÍA

Cd. Mx. Ciudad Universitaria, 2023

1. Datos del alumno
López
Benítez
Raúl
5625871471
Universidad Nacional Autónoma de México
Facultad de Ciencias
Matemáticas Aplicadas
315290435

2. Datos del tutor
Dra.
Ruth Selene
Fuentes
García

3. Datos del sinodal 1
Dra.
Lizbeth
Naranjo
Albarrán

4. Datos del sinodal 2
M. en C.
María Fernanda
Sánchez
Puig

5. Datos del sinodal 3
Dr.
José de Jesus
Galaviz
Casas

6. Datos del sinodal 4
Act.
Yadira
Rivas
Godoy

7. Datos del trabajo escrito
Algoritmos de aprendizaje automático aplicados a la clasificación binaria
86p.
2023

Agradecimientos

Desde hace ya casi un año comencé una serie de cambios en mi vida, tanto físicos como mentales, que me han servido para seguir adelante y no rendirme. Uno de ellos fue el comenzar a hacer ejercicio desde la comodidad de mi casa, siguiendo los vídeos de cierta mujer −a la cual le debo mi avance−. Actualmente las rutinas de ejercicio incluyen dos instrucciones que tienen que ver con el bienestar mental: una es agradecer tres cosas en tu vida tanto al despertar como al acostarse a dormir, y la otra es decir tres cosas positivas sobre ti antes de comenzar tu día. Nunca he sido una persona que tenga un hábito relacionado a tales actividades, sin embargo, estos días he pensado en las cosas por las cuales me puedo sentir agradecido en mi vida, y, afortunadamente, todo lo que ha pasado en mi mente puedo plasmarlo en estas hojas.

Me siento sumamente agradecido con todas las personas que he conocido a través de esta aventura que inició desde hace ya cinco años. Si pudiese exclamar mi gratitud hacia algún ente supremo, le agradecería por las hermosas personas que he conocido en la carrera. No sería nada ni hubiese podido llegar hasta el final sin mi equipo: Itzel, Lázaro, Hendrix, Monse, Magda, Erick, Pablo, Karla, Esme y Esmeralda. Por otro lado, también me siento muy afortunado de haber conocido a personas que se unieron a mi vida en épocas más actuales, y les tengo un amor muy grande: Aldo, Christian, Raúl, José, Miguel, Hugo, Marcos, Daniel, Víctor Faccio y Omar; así como a mis compañeros y amigos del trabajo: Andrea, Belem, Victor, Mariana, Arcelia, Alex y Cynthia.

Agradezco a la vida por haber sido alumno de tan excelentes profesores y profesoras que siempre me alentaron a seguir, me ayudaron a confirmar que mi camino era el correcto y pusieron un poco de su conocimiento en mí: Melissa −quien es una muy querida maestra y amiga−, Rebeca Trejo, León Felipe, Gerardo Avilés, Lizbeth Naranjo, Fernanda Puig, Ruth Selene, Claudia Ivonne, Carlos Barrera, Clara Fittipaldi, Manik Nava, Nery Sofía, Alejandro Cuevas, Yadira Rivas y Giovanni.

Finalmente, pero no menos importante, doy las gracias por todo su apoyo a lo largo de mi vida a mi madre y a mis abuelos Alvina, María y Nicolás. Gracias a Javier, Yoko, Romina y Luis por todo lo que han hecho por mí desde la preparatoria. Gracias a Mariana por siempre estar a mi lado.

Declaración de autenticidad

Por la presente declaro que, salvo cuando se haga referencia específica al trabajo de otras personas, el contenido de esta tesis es original y no se ha presentado total o parcialmente para su consideración para cualquier otro título o grado en esta o cualquier otra Universidad. Esta tesis es resultado de mi propio trabajo y no incluye nada que sea el resultado de algún trabajo realizado en colaboración, salvo que se indique específicamente en el texto.

RAÚL LÓPEZ BENÍTEZ. Ciudad Universitaria, 2023

"It's always darkest before the dawn"

−Florence + The Machine

Dedicado al amor de mi vida, Mariana; a mis hermosos hermosos bebés: Arenita, Misato Katsuragi, Sheldon Belmont; a mis bebés que se fueron a pasear por el universo: Camelia the cat, Strudel On, gatita; a mi mamá, mis abuelos y a mi papá.

Resumen/Abstract

El objetivo del escrito es abordar la teoría y la implementación del aprendizaje supervisado para lograr la clasificación de datos en dos categorías. La primera parte describe las bases teóricas de cada uno de los algoritmos utilizados: regresión logística, clasificador de Bayes, máquinas de soporte vectorial, árboles CART y bosques aleatorios. La segunda parte incluye la implementación de cada uno de los algoritmos en el programa Rstudio, finalizando con la comparación del rendimiento, ventajas y desventajas de cada uno de los métodos. El conjunto de datos utilizado fue tomado del sitio `UCI Machine Learning Repository`[1], que consta de registros de vinos tintos con sus características fisicoquímicas y una variable que indica la calidad del vino, buena o mala calidad.

[1] https://archive.ics.uci.edu/dataset/186/wine+quality

Índice general

Índice de tablas

Introducción

Actualmente los seres humanos vivimos en la era de la información. Día con día se genera una enorme cantidad de datos en prácticamente todos los ámbitos de nuestra vida, principalmente al estar en contacto con los medios digitales, como los teléfonos celulares, las televisiones inteligentes, los ordenadores, entre otros. Escuchar música, disfrutar de películas o series en servicios de *streaming*, realizar búsquedas en internet o utilizar las redes sociales, son algunos ejemplos de acciones que generan, almacenan y comparten enormes conjuntos de datos con los servidores y las empresas detrás de los servicios. Existen algoritmos que aprovechan toda esa información, que no es simplemente almacenada, sino que, se utiliza para hacer un análisis del comportamiento de los individuos y a partir de allí se identifican patrones que funcionan para los fines de las empresas, por ejemplo, ajustar publicidad personalizada o recomendar contenido similar que podría gustarle a un usuario.

El *Machine learning* (traducido como *Aprendizaje de máquina* o *Aprendizaje automático*) es una rama de la Inteligencia Artificial que se encarga de generar algoritmos con la capacidad de aprender y no tener que programarlos explícitamente. Estos algoritmos son alimentados con la mayor cantidad de datos posibles para aprender y conocer todas las distintas situaciones a las cuales podría enfrentarse [1]. El aprendizaje automático se basa en el uso de datos y técnicas matemáticas, probabilísticas, estadísticas y de optimización para lograr que las máquinas aprendan por sí solas. Para llevar a cabo lo mencionado anteriormente es necesario proveer datos, almacenarlos y, finalmente, procesarlos para obtener los resultados deseados.

Existen dos tipos fundamentales de aprendizaje automático: el supervisado y el no supervisado. Únicamente nos centraremos en describir el aprendizaje supervisado en este documento. El aprendizaje supervisado ocurre cuando se entrena al algoritmo dándole las características y etiquetas de los datos explícitamente, así en un futuro se podría hacer una predicción al conocer los atributos de los nuevos datos. Este tipo de aprendizaje se divide en clasificación y regresión:

- **Regresión.** Los algoritmos de regresión son utilizados cuando la variable respuesta es un valor numérico. Para nuevos datos, los algoritmos indicarían una respuesta numérica basada en los atributos conocidos. Los algoritmos más utilizados para la tarea de regresión son los basados en modelos lineales (como la regresión lineal), modelos lineales generalizados (regresión Poisson, regresión Gamma, etc.) y modelos no lineales (como las máquinas de soporte vectorial o los árboles de regresión).

- **Clasificación.** Los algoritmos de clasificación predicen etiquetas categóricas con base en patrones descubiertos a partir de los datos que se les fueron dados. Es decir, para nuevos datos, el algoritmo indicará la clase o categoría a la que deben pertenecer con lo aprendido anteriormente. La variable por predecir puede ser categórica binaria, nominal u ordinal. Los algoritmos más utilizados para la clasificación son los árboles de decisión, la regresión logística, las máquinas de soporte vectorial, los clasificadores de Bayes y los algoritmos basados en redes neuronales.

La clasificación es un proceso conformado fundamentalmente por dos pasos. En el primer paso se construye un clasificador a partir de analizar o aprender de un conjunto de entrenamiento. En el segundo paso el modelo es puesto a prueba con un nuevo conjunto (independiente al conjunto de entrenamiento) para calificar su exactitud y rendimiento; si éstos son aceptables, el clasificador puede utilizarse para cualquier otro conjunto que le sea dado.

La importancia de clasificar radica en que tal acción es una actividad inherente al ser humano; constantemente se busca etiquetar o catalogar todo lo que se conoce para tratar de tener control y atribuir características comunes entre los individuos pertenecientes a ciertos grupos. Por otro lado, la clasificación es fundamental para la toma de decisiones; por ejemplo, saber si un hongo es venenoso o no es primordial para tomar la decisión de ingerirlo.

Aunque dicha tarea puede lograrse utilizando un único algoritmo, es recomendable construir más de uno para poder hacer una comparación de las habilidades y métricas de los modelos, y así, al final hacer un contraste de lo obtenido para elegir el mejor modelo. Para la comparación de los modelos se toma en cuenta los siguientes aspectos:

- Exactitud. Habilidad de predecir correctamente las etiquetas de clase de nuevos datos.
- Velocidad. Costo computacional involucrado en la generación y uso del clasificador.
- Robustez. Habilidad del clasificador de realizar predicciones de manera correcta aun si existen datos con ruido, valores atípicos o valores ausentes.
- Interpretabilidad. Nivel de entendimiento y de visión que es proporcionado por el clasificador. Este aspecto es totalmente subjetivo.

El objetivo de la presente tesis es dar a conocer los algoritmos más utilizados del aprendizaje automático para clasificación binaria; desde la formulación teórica hasta la implementación en el software Rstudio, y posteriormente la evaluación de los modelos con un conjunto de datos reales para entender su funcionamiento, interpretar sus resultados y finalmente comparar y analizar los rendimientos, sus diferencias y similitudes. Los algoritmos que se emplearán serán: regresión logística, clasificador ingenuo de Bayes, máquinas de soporte vectorial, árboles CART y bosques aleatorios.

Los capítulos 1, 2, 3 y 4 se encargan de explicar las bases teóricas de cada uno de los algoritmos mencionados anteriormente. El capítulo 5 establece las métricas que serán utilizadas para la evaluación y comparación de los rendimientos de los algoritmos. El capítulo 6 explica el tema de correlación y multicolinealidad con la finalidad de entender su papel dentro del análisis estadístico y cómo lidiar con los problemas que pueden existir al momento de ajustar modelos. El capítulo 7 tiene como objetivo entender la naturaleza fisicoquímica de las variables registradas, así como mostrar un análisis estadístico descriptivo para entender el comportamiento individual y conjunto de las variables.

El código empleado se encuentra en el siguiente repositorio: https://github.com/RlxOn/Tesis_Clasificacion-binaria

Capítulo 1

Clasificador ingenuo de Bayes

Un fenómeno se define como una cosa inmaterial, hecho o suceso que se manifiesta y puede percibirse a través de los sentidos o del intelecto. En la naturaleza existen dos tipos de fenómenos[1]: los deterministas y los aleatorios. Un *fenómeno determinista* es aquel que al ser ejecutado y repetido, bajo las mismas condiciones, produce el mismo resultado; por ejemplo, el registrar la temperatura a la cual hierve el agua a nivel del mar. Un *fenómeno aleatorio* es aquel que, cuando se ejecuta y se repite bajo las mismas condiciones, el resultado observado no siempre es el mismo y tampoco puede predecirse; por ejemplo, registrar el precio exacto del petróleo a lo largo de un año.

La Teoría de Probabilidad se encarga del estudio de los fenómenos aleatorios. Dicha teoría modela matemáticamente fenómenos de diversas disciplinas del conocimiento humano en donde es preciso incorporar la incertidumbre o el azar como un elemento esencial del modelo [2]. Dependiendo de los métodos utilizados se puede clasificar a la probabilidad en:

- Probabilidad clásica. Se calcula como el cociente de el número de casos favorables del evento entre el número de casos totales.
- Probabilidad frecuentista. Se define como el cociente de el número de ocurrencias del evento entre el número de veces que el experimento fue realizado.
- Probabilidad subjetiva. Dicha probabilidad es estimada por el observador, dependiendo de la información que éste conozca.
- Probabilidad axiomática. Es la probabilidad fundamentada en la teoría de conjuntos y propone las reglas (axiomas) que el cálculo de probabilidades debe satisfacer.

Para el estudio axiomático de los experimentos aleatorios, es necesario contar con tres elementos:

1. **Espacio muestral.** Es el conjunto de todos los posibles resultados del experimento. Se denota generalmente con la letra griega Ω. A cualquier subconjunto del espacio muestral se le llama *evento*.

[1] También llamados *experimentos*.

2. σ**-álgebra** $\mathcal{F}$. Se trata de una familia de subconjuntos del espacio muestral Ω.

3. **Medida de probabilidad.** Función $\mathbb{P}$ en $\mathcal{F}$ que toma valores en el intervalo $[0, 1]$.

Dada la probabilidad axiomática, se pueden efecutar cálculos de probabilidades de conjuntos y de sus operaciones definidas (unión, intersección, diferencia y complemento). Por ejemplo, se puede calcular la probabilidad de obtener un 4 o un 6 al lanzar un dado, la probabilidad de obtener dos águilas al tirar dos monedas, etcétera.

1.1. Probabilidad condicional e independencia

Supongamos que queremos calcular la probabilidad de que la suma de dos dados sea igual a seis si se sabe que el primer dado cayó tres. Como podemos observar, el segundo dado está sujeto al valor que toma el primero, es decir, está condicionado a tomar un cierto conjunto de valores debido a lo que ya se conoce del otro dado.

Si tenemos A, B dos eventos, tal que la probabilidad del evento B es mayor a cero, se define la probabilidad condicional del evento A dado B como la probabilidad de la intersección de los eventos entre la probabilidad del condicionante[2]. En términos matemáticos, esto se expresa como:

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} \tag{1.1}$$

De la ecuación (1.1) puede despejarse a la probabilidad de la intersección, así

$$\mathbb{P}(A \cap B) = \mathbb{P}(A|B)\mathbb{P}(B) \tag{1.2}$$

En consecuencia de la definición de probabilidad condicional, surge un resultado bastante útil y con una amplia aplicación en la teoría de la probabilidad, el llamado Teorema de probabilidad total.

Teorema 1 (Teorema de probabilidad total)**.** *Sea $B_1, ..., B_n$ una partición de Ω tal que $\mathbb{P}(B_i) \neq 0$, con $i = 1, ..., n$. Para cualquier evento A,*

$$\mathbb{P}(A) = \sum_{i=1}^{n} \mathbb{P}(A|B_i)\mathbb{P}(B_i)$$

Ahora, supongamos que una pareja desea tener tres hijos. ¿Cuál es la probabilidad de que el segundo hijo sea mujer?, ¿cuál es la probabilidad de que el segundo hijo sea mujer dado que el primero es mujer? Si efectuamos los cálculos necesarios, obtendremos que la probabilidad de que el segundo hijo sea mujer es 0.5; por otro lado, la probabilidad de que el segundo hijo sea

[2]La probabilidad condicional no está definida para los casos donde la probabilidad del condicionante es igual a cero.

mujer dado que el primero el mujer resulta ser 0.5. En la situación anterior podemos observar que condicionar el evento no cambió en nada la probabilidad, por lo que podemos decir que los eventos son *independientes.*

Decimos que dos eventos A y B son independientes si

$$\mathbb{P}(A|B) = \mathbb{P}(A) \tag{1.3}$$

Si tomamos lo anterior y la ecuación (1.2), de manera equivalente a la definición anterior, decimos que A y B son independientes si y sólo si

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B)$$

Es decir, decimos que dos eventos son independientes si podemos expresar la probabilidad de la intersección como el producto de sus probabilidades.

Las definiciones de probabilidad condicional e independencia se pueden extender de manera muy similar a las variables aleatorias[3]. Si X y Y son variables aleatorias, entonces se define a la probabilidad de X condicionada a Y como

$$\mathbb{P}(X = x|Y = y) = \frac{\mathbb{P}(X = x, Y = y)}{\mathbb{P}(Y = y)}$$

Por otro lado, si X y Y son variables aleatorias con funciones de densidad marginales $f_X(x)$ y $f_Y(y)$ respectivamente, y con función de densidad conjunta[4] $f_{XY}(x, y)$, decimos que X y Y son independientes si

$$f_{X,Y}(x, y) = f_X(x)f_Y(y)$$

La expresión anterior puede extenderse para n variables aleatorias, donde decimos que las variables son independientes si

$$f_{X_1,...,X_n}(x_1, ..., x_n) = \prod_{i=1}^{n} f_{X_i}(x) \tag{1.4}$$

en otras palabras, las variables aleatorias son independientes si se puede expresar a su distribución conjunta como el producto de las distribuciones marginales.

1.2. Teorema de Bayes

El Teorema de Bayes fue publicado por primera vez en 1763, dos años después de la muerte de su creador: el matemático y teólogo inglés Thomas Bayes [2]. Este teorema involucra

[3]Una variable aleatoria X se define como una función del espacio muestral Ω a los números reales ($X : \Omega \to \mathbb{R}$), de tal manera que para todo número real x, $\{\omega \in \Omega : X(\omega) \leq x\} \in \mathcal{F}$.

[4]La función de densidad conjunta de (X, Y) se define como: $f_{X,Y}(x, y) = \mathbb{P}[X = x, Y = y]$. Dada la función de densidad conjunta de (X, Y), se define la función de densidad marginal de X (de forma equivalente con Y) como $\int_{-\infty}^{\infty} f_{X,Y}(x, y)dy$. En caso de que la distribución conjunta sea discreta, se cambia la integral por una suma.

probabilidades condicionales y su demostración incluye al teorema de probabilidad total.

Teorema 2 (Teorema de Bayes)**.** *Sea $B_1, ..., B_n$ una partición de Ω, tal que $\mathbb{P}(B_i) > 0$, $i = 1, ..., n$. Sea A un evento tal que $\mathbb{P}(A) > 0$. Entonces para cada $j = 1, 2, ..., n$,*

$$\mathbb{P}(B_j|A) = \frac{\mathbb{P}(A|B_j)\mathbb{P}(B_j)}{\sum_{i=1}^{n} \mathbb{P}(A|B_i)\mathbb{P}(B_i)}$$

Frecuentemente a la probabilidad $\mathbb{P}(B_j)$ se le llama probabilidad *a priori*, y a la probabilidad resultante $\mathbb{P}(B_j|A)$ se le conoce como probabilidad *a posteriori*, por lo que se dice que el Teorema de Bayes «actualiza» las probabilidades, debido a que pasamos de la probabilidad *a priori* a la probabilidad *a posteriori*. Sin embargo, hay que tener cuidado con $\mathbb{P}(A|B_j)$ y la probabilidad a posteriori, pues, aunque son similares en su escritura, no son las mismas y difieren totalmente en su interpretación. Por ejemplo, no es lo mismo la probabilidad de que una mujer esté embarazada dado que la prueba de embarazo resultó positiva, a la probabilidad de que una prueba de embarazado resulte positiva dado que se la realizó una mujer embarazada.

1.3. Clasificador de Bayes

El Clasificador Ingenuo de Bayes (en inglés *Naive Bayes Classifier*) es una técnica de aprendizaje supervisado para clasificar datos en distintas clases [3]. Si tenemos muestras representativas de distintas clases, el clasificador utilizará el Teorema de Bayes para calcular la probabilidad de que nuevos datos estén en cada clase dada la información que ya conoce. Se le llama «ingenuo» debido a que asume la independencia de las variables predictoras (lo cual no ocurre en la mayoría de los casos).

El Clasificador Ingenuo de Bayes es utilizado en múltiples tareas, por ejemplo:

- Clasificar datos en formato de texto y realizar análisis de sentimiento.
- Detección de spam en correos.
- Catalogar si una noticia es buena o mala.
- Predecir la dirección en la cual *tweets* en Twitter van a influir en una elección.
- Determinar si *tweets* publicados provienen de un bot.

Este clasificador puede explicarse sencillamente con el siguiente ejemplo que utiliza reseñas buenas o malas de libros para predecir si una nueva reseña será buena o mala: Supongamos que tenemos una gran colección de reseñas buenas (C_1) y una gran colección de malas reseñas (C_2). En dichas reseñas aparecen ciertas X_p palabras de nuestro interés. Entonces conocemos la siguiente información:

- La proporción de buenas reseñas.
- La proporción de malas reseñas.
- Las veces que cierta palabra X_i aparece en cualquier reseña (sin importar si es buena o mala).
- El número de veces que cierta palabra X_i aparece en una buena reseña, así como la cantidad de ocasiones que tal palabra aparece en una mala reseña.
- La cantidad de veces que las X_p palabras aparecen en una buena reseña, así como la cantidad de veces que aparecen en una mala reseña.

Por lo tanto, dicha información puede traducirse en probabilidades y así calcular la probabilidad de que una nueva reseña sea buena (o mala) dado las palabras que aparecen en ella. En términos de probabilidad, lo anterior puede expresarse de la siguiente forma:

$$\mathbb{P}(C_1|X_1,...,X_p) = \frac{\mathbb{P}(X_1,...,X_p|C_1)\mathbb{P}(C_1)}{\mathbb{P}(X_1,...,X_p)}$$

Por la suposición de independencia (ecuación (1.4)) del clasificador, entonces

$$\mathbb{P}(C_1|X_1,...,X_p) = \frac{\prod_{i=1}^{p}\mathbb{P}(X_i|C_1)\mathbb{P}(C_1)}{\prod_{i=1}^{p}\mathbb{P}(X_i)}$$

De forma similar para C_2 se tiene

$$\mathbb{P}(C_2|X_1,...,X_p) = \frac{\prod_{i=1}^{p}\mathbb{P}(X_i|C_2)\mathbb{P}(C_2)}{\prod_{i=1}^{p}\mathbb{P}(X_i)}$$

Después de hacer el cálculo de tales probabilidades, el clasificador las compara y así asigna a la nueva reseña en la categoría con mayor probabilidad. Sin embargo, podemos notar que en ambas probabilidades el divisor es exactamente el mismo, por lo cual se puede omitir[5]. Por otro lado, debido a que se efectuan una gran cantidad de productos de números pequeños, usualmente se calcula el logaritmo de las probabilidades. Así el clasificador compara

$$\ln(\mathbb{P}(C_1)) + \ln\left(\sum_{i=1}^{p}\mathbb{P}(X_i|C_1)\right) \quad \text{con} \quad \ln(\mathbb{P}(C_2)) + \ln\left(\sum_{i=1}^{p}\mathbb{P}(X_i|C_2)\right)$$

El procedimiento anterior puede generalizarse para k clases, donde el clasificador asignará los nuevos datos a la clase que mayor probabilidad presente, comparando

$$\ln(\mathbb{P}(C_j)) + \ln\left(\sum_{i=1}^{p}\mathbb{P}(X_i|C_j)\right) \quad \text{con } j = 1,...,k$$

[5]Se asume que la probabilidad de todo X_i es mayor a cero, y por lo tanto $\prod_{i=1}^{p}\mathbb{P}(X_i) > 0$.

Los clasificadores bayesianos pueden ser utilizados para datos continuos (clasificador gaussiano), datos que indican conteo (clasificador multinomial) y datos binarios (clasificador Bernoulli). Los clasificadores multinomiales y Bernoulli cuentan con un único parámetro, alfa, el cual controla la complejidad del modelo. La forma en que alfa funciona es que el algoritmo suma a los datos muchos puntos virtuales que tienen valores positivos para todos los predictores. Esto da como resultado un «suavizamiento» de las estadísticas. Un alfa grande significa más suavizado, lo que resulta en modelos menos complejos. El rendimiento del algoritmo es relativamente robusto para la configuración de alfa, lo que significa que configurar alfa no es fundamental para un buen rendimiento, empero, afinarlo generalmente mejora un poco la precisión.

Para datos de muy alta dimensión (datos con una gran cantidad de variables o número de registros) se emplean principalmente los clasificadores gaussianos, mientras que las otras dos variantes se utilizan ampliamente para datos de recuento escasos, como texto. Los clasificadores multinomiales generalmente se desempeñan mejor que los Bernoulli, en particular para conjuntos de datos con un gran número de características distintas de cero (es decir, documentos grandes) [4].

La rapidez del clasificador de Bayes es alta, incluso puede ser más rápido que los clasificadores empleados por modelos lineales. No obstante, el precio pagado por esta eficiencia es que el clasificador bayesiano frecuentemente proveen de un desempeño ligeramente peor que el de los clasificadores lineales. La razón por la que los clasificadores de Bayes son tan eficientes es que ellos aprenden los parámetros de observar cada característica individualmente y recopilar estadísticas simples por clase de cada una de las características [4].

Estos clasificadores comparten muchas de sus fortalezas y debilidades con los modelos lineales. Son muy rápidos para entrenar y predecir, y el proceso de entrenamiento es fácil de entender. Los modelos funcionan de manera óptima con datos de alta dimensión y son relativamente robustos respecto a los parámetros; además, son excelentes modelos de referencia y se utilizan a menudo en conjuntos de datos muy grandes, donde entrenar incluso a un modelo lineal puede tardar demasiado.

Capítulo 2

Modelos lineales generalizados

En Estadística un análisis de regresión es una técnica que permite modelar la relación entre una variable (llamada variable respuesta o variable dependiente) y otra variable o conjunto de variables (llamadas variables explicativas o variables independientes). El análisis de regresión puede utilizarse para la descripción de los datos, la estimación de parámetros y la predicción y estimación de datos que no pertenecen al conjunto observado.

El procedimiento general para realizar un análisis de regresión es:

1. Obtener los datos.
2. Obtener las estadísticas descriptivas y realizar gráficos que permitan visualizar el comportamiento de los datos, las distribuciones de las variables y su relación con las demás.
3. Proponer y ajustar un modelo.
4. Medir qué tan bueno es el ajuste del modelo (utilizando bondad de ajuste) y verificar el cumplimiento de los supuestos. En caso de contar con más de un modelo, es recomendable comparar las métricas que describen el ajuste para elegir el modelo óptimo.
5. Utilizar el modelo para la predicción, estimación y descripción.

2.1. Modelos Lineales

Un modelo lineal es propuesto cuando se quiere demostrar si existe una relación lineal entre una variable respuesta Y y una variable X o conjunto de variables explicativas X_i. Para cualquier modelo lineal, la variable respuesta es del tipo continua y las variables explicativas pueden ser continuas o categóricas. La fórmula general de un modelo lineal es

$$Y_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_k x_{ik} + \varepsilon_i, \quad i = 1, ..., n$$

donde las β_i son constantes desconocidas llamadas *coeficientes de la regresión*, y ε_i es un error

aleatorio y desconocido [5].

El modelo se puede expresar de manera matricial como

$$\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ \vdots & & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{nk} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

y a su vez, se formula de manera contracta mediante la ecuación

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon \tag{2.1}$$

donde a la matrix $\mathbf{X}$ se le llama *matriz diseño* y $\varepsilon \sim N_n(0, \sigma^2 I_n)$, o de manera equivalente, $\mathbf{Y} \sim N_n(\mathbf{X}\beta, \sigma^2 I_n)$.

El conjunto de variables explicativas X_{ik} siempre es controlado, por su parte, la variable Y_i es aleatoria. Por lo tanto, existe una distribución de probabilidad para Y_i en cada valor de las X_i, que, como puede verse en la ecuación (2.1), se trata de una distribución normal multivariada. La media de la distribución de cada Y_i está dada por $\mathbf{X}\beta$, lo cual puede expresarse como

$$\mathbb{E}[Y_i | X_{i1}, ..., X_{ik}] = \beta_0 + \beta_1 x_{1k} + ... + \beta_k x_{ik} \tag{2.2}$$

en otras palabras, existe una relación lineal entre el valor esperado de cada Y_i y las covariables X_{ik}.

Los coeficientes de la regresión se estiman bajo el método de mínimos cuadrados y su interpretación es muy simple. β_0 es el intercepto del hiperplano; si el conjunto de valores de las X_i incluyen al cero, entonces $Y = \beta_0$ cuando todas las covariables son iguales a cero. Los parámetros β_i se interpretan como el cambio producido en Y por el cambio en una unidad de X_i (después de controlarse las demás covariables X_j). Si algún coeficiente β_i es estadísticamente igual a cero, esto indica que no se produce cambio en Y por cambio en una unidad de la covariable X_i en cuestión, por lo cual tal X_i no sirve para explicar a la variable respuesta y sería apropiado removerla del conjunto de covariables.

Cuando se tiene únicamente una variable explicativa, el modelo de regresión se reduce a $Y = \beta_0 + \beta_1 X + \varepsilon$, y se le llama *modelo de regresión lineal simple*. En la Figura 2.1 se utilizan la regresión lineal simple para ajustar una recta. Puede interpretarse geométricamente a β_1 como la pendiente de ésta.

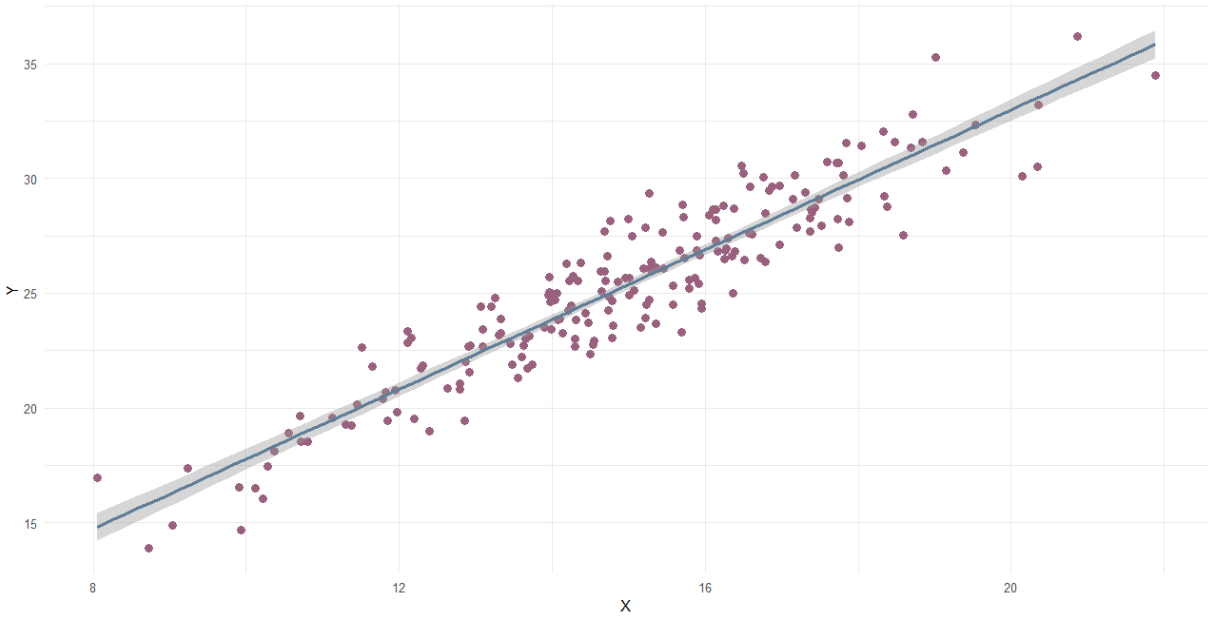

Fig. 2.1: Ejemplo gráfico de un ajuste de regresión lineal simple.

Los modelos lineales, además de cumplir la relación lineal entre la variable respuesta y las covariables explicativas, deben cumplir tres supuestos adicionales respecto a sus residuales[1]:

- Homocedasticidad[2]. La varianza debe ser igual para todos los residuales.
- No correlación. La correlación entre los residuales debe ser nula.
- Normalidad. Los residuales deben seguir una distribución normal.

Cuando alguno de los supuestos no se cumple, es recomendable realizar transformaciones a la variable respuesta, a las covariables explicativas o a todo el conjunto de datos para garantizar, en este nuevo modelo transformado, el cumplimiento de los supuestos de la regresión.

2.2. Modelos lineales generalizados

Como se mencionó anteriormente, la principal razón por la cual se busca ajustar un modelo de regresión es para predecir una variable de interés dado un conjunto de covariables explicativas. Sin embargo, el modelo de regresión lineal funciona únicamente cuando la variable respuesta es del tipo continua; si quisiéramos ajustar un modelo de regresión donde la variable respuesta fuese discreta o categórica, nos encontraríamos con distintas cuestiones. Por ejemplo, supongamos que

[1]Para un modelo de regresión lineal los residuales se definen como los valores observados menos los valores estimados, es decir, $e_i = y_i - \hat{y}_i$, $i = 1, ..., n$.

[2]También conocida como varianza constante.

queremos modelar la probabilidad de que ocurra un suceso (nombrémoslo $Y = 1$) con un ajuste lineal. La ecuación obtenida sería la siguiente:

$$\mathbb{P}(Y = 1) = \beta_0 + \beta_1 X_1 + ... + \beta_k X_k$$

Si estimamos los parámetros y ajustamos el modelo lineal, tendríamos fundamentalmente dos problemas:

- Los valores predichos de la probabilidad de ocurrencia del suceso podrían estar fuera del intervalo $[0, 1]$.
- Los intervalos de confianza y las pruebas de significancia de los coeficientes de la regresión suponen que los datos vienen de una distribución normal (o distribuciones muestrales relacionadas con ella), lo cual no ocurre si la variable respuesta es binaria.

Los modelos lineales generalizados (abreviados comúnmente como *GLM*, por sus siglas en inglés) establecen una relación lineal, no entre el valor esperado de la variable respuesta y los predictores (como se ve en la ecuación (2.2)), sino entre una función de la media de la variable respuesta y los predictores, es decir

$$g(\mathbb{E}[Y|X_{i1}, ..., X_{ik}]) = \beta_0 + \beta_1 X_{i1} + ... + \beta_k X_{ik}$$

donde la función $g(\cdot)$ depende de la distribución de la variable respuesta Y[6].

Los modelos lineales generalizados constan de tres componentes:

1. **Componente aleatoria.** Se refiere a la variable respuesta Y. En términos generales, podemos ver a Y como un vector $Y = (y_1, ..., y_n)$ donde las y_i son variables aleatorias independientes e idénticamente distribuidas, además de pertenecer a la *familia exponencial*, lo que significa que la función de probabilidad de cada y_i puede expresarse de la forma

$$f(y; \theta) = s(y)t(\theta) \exp[a(y)b(\theta)] \tag{2.3}$$

donde $a(y)$, $b(\theta)$, $s(y)$ y $t(\theta)$ son funciones conocidas y distintas de cero, además de que a y b son no constantes. Podemos reescribir a la ecuación (2.3) como

$$f(y; \theta) = \exp[a(y)b(\theta) + c(\theta) + d(y)] \tag{2.4}$$

donde $s(y) = \exp[d(y)]$ y $t(\theta) = \exp[c(\theta)]$. Si $a(y) = y$ se dice que la distribución es de *forma canónica*, y en algunas ocasiones a $b(\theta)$ se le llama *parámetro natural* de la distribución. Si existen otros parámetros, además del parámetro de interés θ, se consideran como parámetros de perturbación que forman parte de las funciones a, b, c y d y generalmente se consideran conocidos [7].

Pertenencen a la familia exponencial la distribución Binomial, Bernoulli[3], Poisson, Binomial

[3]Como caso particular de una binomial con parámetros $(1, p)$

negativa, Geométrica[4], Normal, Exponencial[5], Gamma, Beta, entre otras.

2. **Componente sistemática.** El conjunto de covariables $X_1, ..., X_k$ forma un predictor lineal dado por

$$\eta = \sum_{j=1}^{k} x_j \beta_j = \mathbf{X}\beta$$

donde $x_j = (x_{1j}, ..., x_{nj})$ y x_{ij} es el valor de la j-ésima covariable para la observación i; η es un vector de dimensión $n \times 1$; $\mathbf{X}$ es la matriz diseño de dimensión $n \times k$; y $\beta = (\beta_1, ..., \beta_k)'$ es un vector de dimensión $k \times 1$ de parámetros desconocidos que se estimarán para ajustar el modelo de regresión [8].

3. **Función liga.**[6] Función que relaciona la media de Y con las variables predictoras X_j. La *función liga canónica* es la función que transforma la media de Y en el parámetro θ. En otras palabras, si $g(\cdot)$ es una función liga canónica, ocurre que $g(\mathbb{E}[Y]) = \theta$.

La Tabla 2.1 muestra las distribuciones y las funciones liga canónicas utilizadas para los modelos lineales generalizados más comunes.

Distribución de Y	**Función liga canónica**
Normal	$\mathbf{X}\beta = \mathbb{E}[Y]$ (identidad)
Poisson	$\mathbf{X}\beta = \ln(\mathbb{E}[Y])$ (logarítmica)
Binomial	$\mathbf{X}\beta = \ln\left(\frac{p}{1-p}\right)$ (logística)
Gamma	$\mathbf{X}\beta = \frac{1}{\mathbb{E}[Y]}$ (recíproca)

Tabla 2.1: Distribuciones utilizadas para los GLM y sus funciones liga canónicas.

2.3. Regresión logística

En la regresión logística la variable respuesta Y es binaria, es decir, Y solo toma como valores a cero (fracaso) y a uno (éxito). Si p es la probabilidad de que $Y = 1$ (y por consiguiente la probabilidad de que $Y = 0$ es $1 - p$), entonces Y es una variable aleatoria con distribución Bernoulli de parámetro p.

$$Y \sim Ber(p)$$
$$f(y) = p^y(1-p)^{1-y} \quad y = 0, 1$$

[4]Caso particular de una binomial negativa con parámetros $(1, \theta)$
[5]Caso particular de una Gamma con parámetros $(1, \lambda)$.
[6]También llamada función de enlace o función *link*.

Dado lo anterior, lo que se desea estimar en un modelo de regresión logística es la probabilidad de ocurrencia p del evento de interés. Sin embargo, como se mencionó anteriormente, al trabajar con probabilidades podemos enfrentar el problema de salir del intervalo $[0, 1]$ si las relacionamos directamente con los predictores. La solución sencilla es aplicar transformaciones a la probabilidad y relacionarla así con los predictores para evitar tal problemática. La transformación utilizada para la regresión logística recibe el nombre de función *logit* o *log-odds*; dicha función relaciona de manera lineal a los predictores con el logaritmo natural de los momios.

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + ... + \beta_k X_k \tag{2.5}$$

La función logit está acotada en el intervalo [0,1]; se aproxima a $-\infty$ cuando la probabilidad tiende a cero y se aproxima a $+\infty$ cuando la probabilidad tiende a 1.

Los *momios*(del inglés *odds*) se definen como el cociente de la probabilidad de ocurrencia de un evento entre la probabilidad de no-ocurrencia, e indican proporcionalmente cuánto más probable es la ocurrencia del evento comparada con la no-ocurrencia. Los momios no deben interpretarse naturalmente como una probabilidad, sino una proporción. Si los momios son menores a uno, es más probable la no-ocurrencia del evento; si los momios son iguales a uno, la ocurrencia y no-ocurrencia son equiprobables; y si los momios son mayores a uno, es más probable la ocurrencia del evento. Como puede verse en la Figura 2.2, los momios toman valores desde cero (cuando $p = 0$) hasta infinito (cuando $p = 1$).

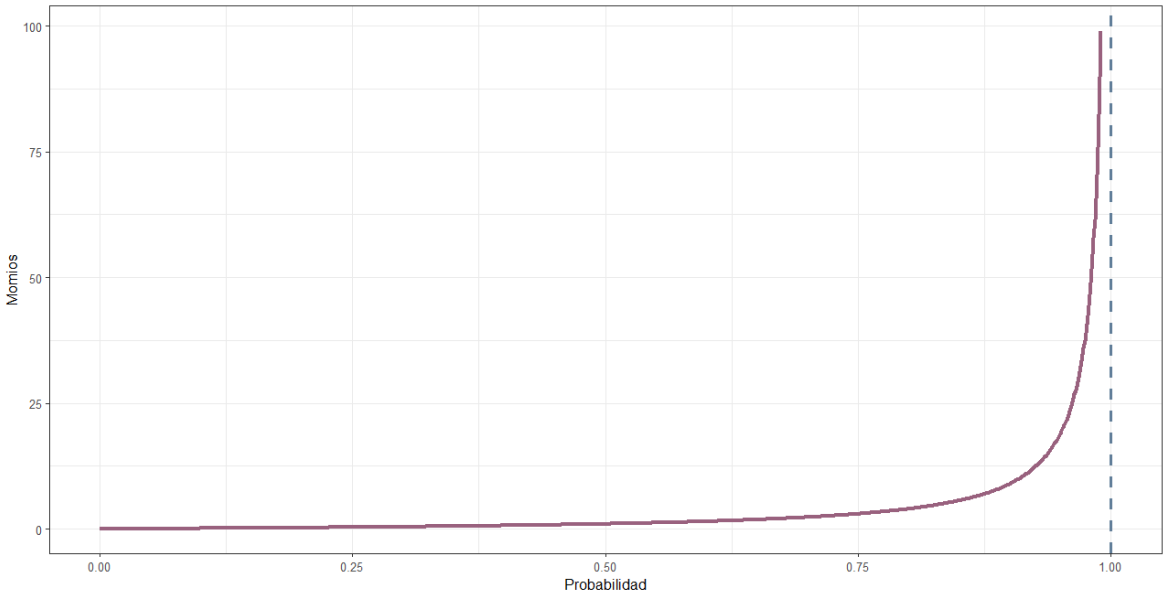

Fig. 2.2: Gráfica de la probabilidad p contra los momios.

Para obtener a la probabilidad estimada dados los predictores, debemos despejar a p de la

ecuación (2.5). Aplicamos la función exponencial, así

$$\frac{p}{1-p} = \exp[\beta_0 + \beta_1 X_1 + ... + \beta_k X_k] \tag{2.6}$$

Finalmente, utilizando álgebra, obtenemos

$$p = \frac{\exp[\beta_0 + \beta_1 X_1 + ... + \beta_k X_k]}{1 + \exp[\beta_0 + \beta_1 X_1 + ... + \beta_k X_k]} \tag{2.7}$$

Los parámetros β_i se estiman maximizando la función de verosimilitud que representa la verosimilitud de que los datos observados sean una muestra de una variable con una determinada distribución, con lo cual lo que se está haciendo es calcular los valores de los parámetros que hacen más verosímiles los datos. Como maximizar la verosimilitud equivale a maximizar el logaritmo natural de ésta, se utiliza la ecuación

$$\ln(L(\beta)) = \sum y_i \ln(p_i) + (n_i - y_i) \ln(1 - p_i)$$

donde se escribe a p_i en función de β y se maximiza esa función; para ello se debe resolver una serie de ecuaciones que carecen de solución analítica, debido a esto, se utiliza un método iterativo basado en el algoritmo de *Newton-Raphson* [6].

La interpretación de los coeficientes de la regresión pueden interpretarse con respecto a la función logit o con respecto a los momios. Si nos fijamos en la función logit, la interpretación de los coeficientes es muy similar al modelo de regresión lineal; β_0 indica el valor que tomaría la función logit si todas las covariables fueran iguales a cero, por su parte, β_i indica el cambio producido en la función logit por el cambio en una unidad de la variable X_i (después de controlar las demás covariables X_j).

Si nos fijamos en los momios, la interpretación es de la siguiente forma: supongamos que X_i se incrementa en una unidad (y controlamos las demás covariables X_j), entonces

$$\frac{p}{1-p} = \exp[\beta_0 + \beta_i(X_i + 1)] = \exp[\beta_0 + \beta_i X_i] \exp[\beta_i]$$

en otras palabras, los momios se multiplicarían por $\exp[\beta_i]$. Si β_i es pequeña entonces $(\exp[\beta_i] - 1) \approx \beta_i$, y en este caso $100\beta_i$ es el cambio porcentual aproximando en los momios cuando X_i se incrementa una unidad.

Si $\beta_i < 0$, el efecto de incremento en X_i implicará que decrezcan los momios. Si $\beta_i > 0$ entonces los momios incrementarán [8]. Visto desde el punto de vista de la probabilidad, si $\beta_i < 0$, al decrecer los momios, significa que se hace más probable la no-ocurrencia del suceso; y si $\beta_i > 0$, se hace más probable la ocurrencia del suceso.

Si $\beta_i = 0$, entonces $\exp[\beta_i] = 1$, por lo cual no existe efecto en los momios. Desde el punto

de vista de la probabilidad, si $\beta_i = 0$, esto indicaría que $p = \frac{1}{2}$, lo cual no brinda información adicional sobre la ocurrencia del suceso. En caso de que, al estimar los coeficientes, algún β_i sea estadísticamente igual a cero, esto indicaría que la variable X_i en cuestión no produce cambio en los momios y por lo tanto no brinda información adicional para la regresión, razón por la cuál debería considerarse removerla del conjunto de covariables.

Cuando solo existe una variable explicativa, la ecuación (2.5) se reduce a

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X$$

y de manera análoga, la ecuación (2.7) se reduce a

$$p = \frac{\exp[\beta_0 + \beta_1 X)]}{1 + \exp[\beta_0 + \beta_1 X]}$$

ésta última se puede graficar dados los valores estimados de β_0 y β_1. Si $\beta_1 > 0$, la curva será creciente, lo cual indica que el incremento en X aumenta la probabilidad de ocurrencia. Si $\beta_1 < 0$, la curva será decreciente, lo cual indica que el incremento en X aumenta la probabilidad de no-ocurrencia (véase Figura 2.3).

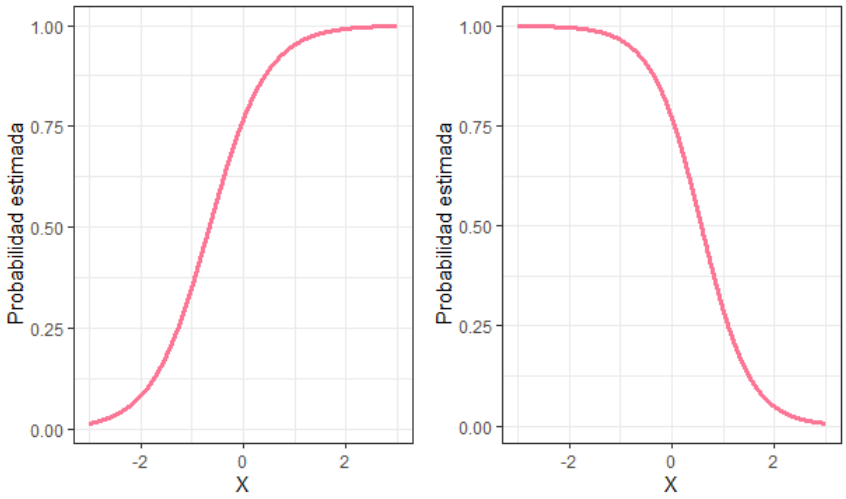

Fig. 2.3: Gráficas de X contra la probabilidad estimada. Si $\beta_1 > 0$, la curva ajustada será creciente (izquierda), mientras que si $\beta_1 < 0$, la curva ajustada será decreciente (derecha).

A diferencia de la regresión lineal, la regresión logística no requiere de homocedasticidad, no correlación y normalidad respecto a los residuales. Para la regresión logística los supuestos del modelo son los siguientes [9]:

- Respuesta binaria. La variable respuesta sólo toma dos valores.

- Baja multicolinealidad. Se requiere de baja o nula multicolinealidad[7] entre los predictores (en caso de ser más de uno).
- Linealidad. Debe existir una relación lineal entre la variable respuesta y la función logit.

[7] Se explicará la multicolinealidad en el capítulo 6.

Capítulo 3

Máquinas de Soporte Vectorial

Las máquinas de soporte vectorial (*Support Vector Machines*, y abreviadas como SVM en inglés) se definen como un conjunto de algoritmos de aprendizaje supervisado pertenecientes a la familia de los clasificadores lineales desarrollados por Vladimir Vapnik y su equipo en los laboratorios AT&T alrededor del año 1995 [10].Este conjunto de algoritmos utiliza un espacio de hipótesis de funciones lineales en un espacio de mayor dimensión inducido por un Kernel, en el cual las hipótesis son entrenadas por un algoritmo tomado de la teoría de optimización [11].

Supongamos que queremos clasificar un conjunto de datos en únicamente dos categorías, una SVM construye un hiperplano, a partir de *vectores de soporte*, en un espacio de dimensión muy alta o incluso infinita. Este hiperplano separa de forma óptima los datos en cada una de las categorías. Las SVM se centran fundamentalmente –hablando de los problemas de clasificación– en el concepto de «separación óptima»; las máquinas buscan el hiperplano que tenga la máxima distancia (margen) con los puntos que estén más cerca de él mismo. Por esta razón a veces también se les conoce como «clasificadores de margen máximo».

Las máquinas de soporte vectorial han ganado gran popularidad como herramienta para la identificación de sistemas no lineales debido a que principalmente éstas se encuentran basadas en el principio de minimización del riesgo estructural (*Structural Risk Minimization*, SRM por su abreviatura en inglés), originado de la teoría de aprendizaje estadístico desarrollada por Vapnik en su libro *The nature of statistical learning theory* [12]. Dicho principio ha demostrado ser superior al principio de minimización del riesgo empírico (*Empirical Risk Minimization*, ERM por su abreviatura en inglés), utilizado por las redes neuronales convencionales.

Entre las múltiples aplicaciones que las máquinas de soporte vectorial tienen, se encuentra el reconocimiento de dígitos escritos a mano, reconocimiento de objetos, identificación de voz, detección de intrusos, autenticación de rostros, detección de pupilas y clasificación del texto [13]. En el caso de la estadística y el aprendizaje automático, las máquinas de soporte vectorial pueden ser utilizadas para clasificación binaria, clasificación multinomial y para ajustes de regresión.

Las ventajas y fortalezas que ofrece este conjunto de algoritmos son:

- El entrenamiento es relativamente fácil.

- No se habla de un mínimo local; la estimación de parámetros se realiza a través de optimizar una función de costo convexa.
- Se escalan considerablemente bien para datos en espacios con dimensiones grandes.
- El compromiso entre la complejidad del clasificador y el error peude ser controlado explícitamente.
- Excelente capacidad de generalización, debido a la minimización del riesgo estructurado.
- Existen pocos parámetros a ajustar; el modelo solo depende de los datos con mayor información.
- Datos no tradicionales como cadenas de caracteres y árboles pueden ser usados como entrada a las máquinas, en vez de vectores de características [14].
- El modelo final puede ser escrito como una combinación de un número pequeño de vectores de entrada.
- La solución de SVM es *sparse*, es decir, la mayoría de las variables son cero en la solución de SVM. El modelo final puede ser escrito como una combinación de un número muy pequeño de vectores de entrada, llamados vectores de soporte [11].

Aun si las SVM tienen múltiples ventajas y fortalezas, no están exentas de tener debilidades y desventajas, las cuales son:

- Es necesaria una «buena» función kernel, en otras palabras, se necesita elegir una función kernel óptima para el modelo y se necesitan metodologías eficientes para sintonizar los parámetros de inicialización.
- Pueden ser considerablemente lentas en la fase de prueba.
- No existen métodos exactos que permitan conocer los parámetros de las máquinas, por lo que se debe recurrir a técnicas de evaluación de modelos; esto puede tener un margen de error considerable para encontrar los parámetros óptimos.

3.1. Clasificación para conjuntos linealmente separables

Consideremos un conjunto de elementos representados por el par (x_i, y_i) que llamaremos *tuplas*, cada una con una serie de características que se almacenan en un vector x_i y una categoría asignada que se almacena en la componente y_i. Si representamos al conjunto de tuplas en un espacio y podemos separarlas a través de un hiperplano, de tal forma que cada tupla esté en el semiplano de la clase correcta, entonces se dice que el conjunto de tuplas es *separable* (véase Fig. 3.1).

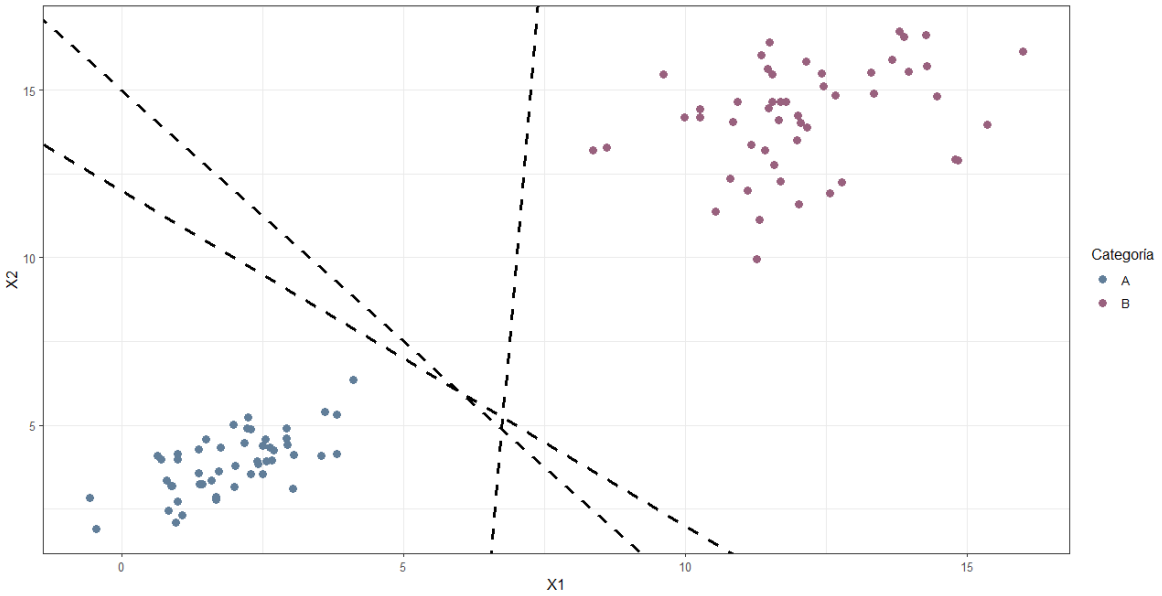

Fig. 3.1: Conjunto de tuplas separable. En este caso vemos múltiples rectas que podrían separar a los datos en la clase correcta.

Dado un conjunto separable $S = \{(x_1, y_1), ..., (x_n, y_n)\}$, donde toda $x_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$, se define un *hiperplano de separación* como una función lineal que separa dicho conjunto sin tener errores de clasificación. Dicho hiperplano se expresa de la siguiente forma:

$$D(x_i) = w_1 x_1 + ... + w_d x_d + b = \langle w, x_i \rangle + b \tag{3.1}$$

donde $w \in \mathbb{R}^d$ y b es un número real cualquiera. Las restricciones que debe satisfacer un hiperplano de separación para toda tupla del conjunto S son:

$$\begin{aligned} \langle w, x_i \rangle + b \geq 0 \quad si \quad y_i = 1 \\ \langle w, x_i \rangle + b \leq 0 \quad si \quad y_i = -1 \end{aligned}$$

O de manera equivalente,

$$y_i D(X_i) \geq 0 \qquad i = 1, ..., n \tag{3.2}$$

No obstante puede existir una cantidad infinita de hiperplanos que satisfagan la restricción anterior. Por consecuente se debe buscar un criterio que proporcione una regla de decisión sobre el conjunto de tuplas, de tal forma que se halle el hiperplano de separación óptimo.

Se define al margen τ como la mínima distancia entre el hiperplano de separación y la tupla más próxima a él de cualquiera de las dos clases. Vapnik y Lerner [15] propusieron en 1963 tomar como hiperplano separador óptimo aquel que maximice el margen. Un hiperplano

separador se dirá óptimo si y sólo si equidista de la tupla más cercana de cada clase[1].

A partir del margen τ se puede formular una nueva restricción con respecto al hiperplano separador: la distancia entre un hiperplano separador $D(x)$ y una tupla x_0 está dada por

$$\frac{|D(x_0)|}{\|w\|}$$

que junto con la ecuación (3.2) establece que todas las tuplas de entrenamiento deben cumplir

$$\frac{y_i D(x_i)}{\|w\|} \geq \tau \qquad\qquad i = 1, ..., n \tag{3.3}$$

De la expresión anterior podemos deducir que encontrar el hiperplano de separación óptimo equivale a hallar a w que maximiza el margen. Ya que existen infinitas soluciones que difieren únicamente en la escala de w, se establece por convención que $\tau\|w\| = 1$. Así concluimos que maximizar el margen equivale a disminuir la norma de w y por lo tanto la desigualdad (3.3) queda como

$$y_i D(x_i) \geq 1 \qquad\qquad i = 1, ..., n \tag{3.4}$$

La búsqueda del hiperplano de separación óptimo se formaliza como un problema de optimización de tipo cuadrático donde buscamos minimizar la norma al cuadrado de w, es decir,

$$\begin{aligned} \text{mín} \quad & f(w) = \tfrac{1}{2}\|w\|^2 = \tfrac{1}{2}\langle w, w\rangle \\ s.a \quad & y_i(\langle w, x_i\rangle + b) - 1 \geq 0 \qquad i = 1, ..., n \end{aligned} \tag{3.5}$$

El hiperplano definido por w y b que minimizan el problema (3.5) recibe el nombre de *hiperplano de separación de margen duro*. Este problema es difícil de resolver debido a la complejidad de las restricciones, por lo que se acude a la teoría de optimización.

Un problema de optimización primal tiene una forma dual si la función a optimizar y sus restricciones son estrictamente convexas. En este caso resolver el problema dual equivale a obtener la solución del primal. Como este problema tiene una función objetivo y restricciones estrictamente convexas[2], se admite un dual. Para hallarlo se hace uso de la función de Lagrange

[1] Esta proposición puede demostrarse por contradicción.

[2] Se dice que una función f es estrictamente convexa en un conjunto S si $f(\lambda x_1 + (1-\lambda)x_2) < \lambda f(x_1) + (1-\lambda)f(x_2)$ para todo $\lambda \in [0, 1]$ y para todo $x_1, x_2 \in S$, con $x_1 \neq x_2$.

$$L(w,b,\alpha) = \frac{1}{2}\langle w,w\rangle - \sum_{i=1}^{n} \alpha_i(y_i(\langle w,x_i\rangle + b) - 1) \qquad i = 1,...,n \tag{3.6}$$

donde α_i son los multiplicadores de Lagrange. Podemos aplicar las condiciones de Karush-Kuhn-Tucker [16], derivando la ecuación respecto a las variables sobre las que optimizamos en el primal e igualando a cero los productos por los multiplicadores de Lagrange obtenemos:

$$\frac{\partial L(w^*,b^*,\alpha)}{\partial w} = w^* - \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \qquad i = 1,...,n$$

$$\frac{\partial L(w^*,b^*,\alpha)}{\partial b} = \sum_{i=1}^{n} \alpha_i y_i = 0 \qquad i = 1,...,n \tag{3.7}$$

$$\alpha_i((1 - y_i(\langle w^*,x_i\rangle + b^*)) = 0 \qquad i = 1,...,n \tag{3.8}$$

De (3.7) obtenemos la expresión de w^* en términos de los multiplicadores de Lagrange y sus restricciones, así

$$w^* = \sum_{i=1}^{n} a_i^* y_i x_i \qquad i = 1,...,n \tag{3.9}$$

$$\sum_{i=1}^{n} a_i^* y_i = 0 \qquad i = 1,...,n \tag{3.10}$$

Si aplicamos las ecuaciones (3.9) y (3.10) a la igualdad (3.1), obtenemos la función a maximizar en el problema dual, el cual, si añadimos las restricciones de no negatividad asociadas a los multiplicadores de Lagrange, es el siguiente:

$$\begin{aligned} \text{máx} \quad & L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j(\langle x_i,x_j\rangle) \\ s.a \quad & \sum_{i=1}^{n} \alpha_i y_i \geq 0, \qquad \alpha_i \geq 0, i = 1,...,n \end{aligned} \tag{3.11}$$

La principal ventaja de resolver el problema dual yace en que el costo computacional es mucho menor, esto se debe a que en el problema dual el número de variables es directamente proporcional al tamaño de la muestra, en cambio el problema primal lo es con la dimensionalidad de las tuplas.

El vector direccional w^* del hiperplano óptimo se obtiene sustituyendo la solución del dual en (3.11)

$$D(x) = \sum_{i=1}^{n} \alpha_i^* y_i(\langle x,x_i\rangle) + b^*, \qquad i = 1,...,n \tag{3.12}$$

Una tupla se dirá *separable* si satisface la restricción (3.4). Se le llamará *vector de soporte* a aquellas tuplas que satisfacen la restricción (3.4) con igualdad exacta. Podemos caracterizar a dichas tuplas utilizando la condición complementaria de KKT (ecuación (3.8)); se deduce que si en una tupla $\alpha_i > 0$ entonces $y_i(\langle w^*, x_i\rangle + b^*) = 1$, por consiguiente se puede afirmar que solo las tuplas que tengan asociado un $\alpha_i > 0$ serán vectores de soporte y así son las únicas tuplas que intervienen en la construcción del hiperplano.

La determinación de b^* se realiza a partir de la ecuación (3.8). Si $\alpha_i > 0$, nos encontramos con un vector de soporte, y en ese caso

$$y_i(\langle w*, x_i\rangle + b^*) = 1$$

Si despejamos a b^*, obtenemos

$$b^* = y_{vs} - \langle w^*, x_{vs}\rangle$$

donde (x_{vs}, y_{vs}) representa cualquier tupla que satisfaga la igualdad anterior, es decir, es un vector de soporte. En la práctica es más robusto obtener b^* haciendo el promedio de todos los vectores de soporte. Sea V el conjunto de los índices de los vectores de soporte y N_v su cardinalidad, b^* está dado por la siguiente expresión:

$$b^* = \frac{1}{N_v}\sum_{i\in V}(y_i - \langle w^*, x_i\rangle)$$

3.2. Clasificación para conjuntos cuasi-linealmente separables

En la práctica es muy frecuente trabajar con conjuntos de tuplas que no son linealmente separables a la perfección. Corinna Cortes y Vladimir Vapnik demuestran en su artículo "Support-vector networks" [17] que los problemas de conjuntos no linealmente separables pueden ser tratados de mejor manera si se permite que algunas tuplas no verifiquen las restricciones sobre el margen en el problema primal. En otras palabras, podemos relajar el grado de separabilidad entre los conjuntos de tuplas permitiendo que existan errores de clasificación para algunas tuplas de entrenamiento. Bajo tales circunstancias puede suceder que una tupla quede dentro del margen asociado a la clase correcta, de acuerdo a la frontera de decisión que define el hiperplano de separación, y en otro caso, la tupla cae al otro lado del hiperplano, quedando clasificada de forma incorrecta. En ambos casos se dice que la tupla es *no separable*.

Para abordar este problema se introduce un conjunto de variables reales no negativas γ_i, $i = 1, ..., n$, llamadas *variables de holgura*, cada una de ellas asociada a una tupla, de tal forma que se puede controlar el número de tuplas no separables. Las restricciones relajadas que deberá verificar cada tupla de entrenamiento son

$$y_i(\langle w, x_i\rangle + b^*) \geq 1 - \gamma_i, \qquad i = 1, ..., n \tag{3.13}$$

Se puede entender a la variable de holgura asociada a cada tupla (x_i, y_i) como la desviación

de la situación separable, medida desde el margen de la clase correspondiente a dicha tupla. Estas variables representan el potencial incumplimiento de las restricciones sobre el margen para cada una de las tuplas. Se pueden clasificar a las tuplas en separables si su variable de holgura correspondiente toma valor cero, no separables pero correctamente clasificadas si su variable de holgura toma valor entre cero y uno, y no separables y mal clasificadas si el valor de su variable es mayor a uno. Por lo tanto, la suma de las γ_i permite medir el costo asociado al número de tuplas no separables; cuanto mayor sea el valor de la suma, mayor será el número de tuplas no separables.

Introducidas las variables de holgura, la función a optimizar debe incluir los errores de clasificación que comete el hiperplano de separación, en otras palabras,

$$f(w, \gamma) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\gamma_i \tag{3.14}$$

donde C es una constante considerablemente grande, cuyo valor se puede determinar manualmente y ésta permite controlar en qué grado influye el término del costo de tuplas no separables en la minimización de la norma, es decir, permite regular el compromiso entre el grado de sobreajuste del clasificador final y la proporción del número de tuplas no separables (complejidad y error de entrenamiento). La elección del valor de C determinará en cierto modo la calidad del clasificador. En consecuencia se tiene que el nuevo problema de optimización es:

$$\begin{aligned} \text{mín} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\gamma_i \\ s.a \quad & y_i(\langle w, x_i\rangle + b^*) + \gamma_i - 1 \leq 0, \\ & \gamma_i \geq 0, \qquad i = 1, ..., n \end{aligned} \tag{3.15}$$

El hiperplano de separación definido a partir de w y b, resultante de resolver el problema (3.15) se denomina *hiperplano de separación de margen blando.*

Vapnik y Chervonenkis [18] demostraron que el aumento de la complejidad de la función de separación provoca una disminución del error de clasificación del conjunto de entrenamiento, sin embargo, esto implica un sobre-entrenamiento de la SVM, de forma que aumenta el riesgo de error cuando se aplica sobre un conjunto de prueba; en otras palabras, la SVM pierde capacidad de generalización. Podemos ver la expresión a minimizar en (3.15) como un compromiso entre la complejidad de la SVM inversamente proporcional a la norma de w y el sobre-entrenamiento controlado por la suma de las variables de holgura.

La resolución del problema primal (3.15) es similar al caso separable de la sección anterior, con la principal diferencia de que al tener dos familias de restricciones en este problema, aparecerán dos familias de multiplicadores de Lagrange. De igual forma, como en el problema anterior, hacemos

uso de la teoría de optimización para así obtener el problema dual:

$$\begin{aligned} \text{máx} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \\ s.a \quad & \sum_{i=1}^{n} \alpha_i y_i = 0, \\ & C \geq \alpha_i \geq 0, \quad i = 1, ..., n \end{aligned} \tag{3.16}$$

Podemos observar que ahora existe una clasificación todavía más completa, que en el caso separable, de los vectores de soporte dado que los multiplicadores de Lagrange α_i tienen a C como cota superior.

3.3. Clasificación para conjuntos no separables linealmente

Cuando el conjunto de tuplas no puede separarse por medio de una función lineal (un hiperplano separador), no queda otra opción más que recurrir a una ténica que consiste en la transformación del espacio original mediante una función no lineal hacia un espacio de Hilbert[3] dotado de un producto escalar denominado *función kernel*. Llamaremos *espacio de entradas* al espacio original de las tuplas x. Al nuevo espacio transformado le llamaremos *espacio de características* y se definirá a partir de un conjunto de funciones base no lineales.

Sea $\phi : \mathbb{X} \to \mathcal{F}$ la transformación que hace corresponder a cada vector de entrada x con un punto en el espacio de características $\mathcal{F}$, donde $\phi(x) = [\phi_1(x), ..., \phi_m(x)]$ y existe $\phi_i(x)$, con $i = 1, ..., m$ tal que cada $\phi_i(x)$ es una función no lineal. Por definición una función *kernel* es una transformación $K : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ que a cada par de elementos del espacio $\mathbb{X} \times \mathbb{X}$ le asigna un valor real correspondiente al producto escalar de las imágenes de dichos elementos en el espacio de características $\mathcal{F}$:

$$K(x, x') = \langle \phi(x), \phi(x') \rangle = \sum_{i=1}^{m} \phi_i(x) \phi_i(x') \tag{3.17}$$

[3]Un Espacio de Hilbert es un espacio vectorial H con un producto escalar que es completo respecto a la norma inducida.

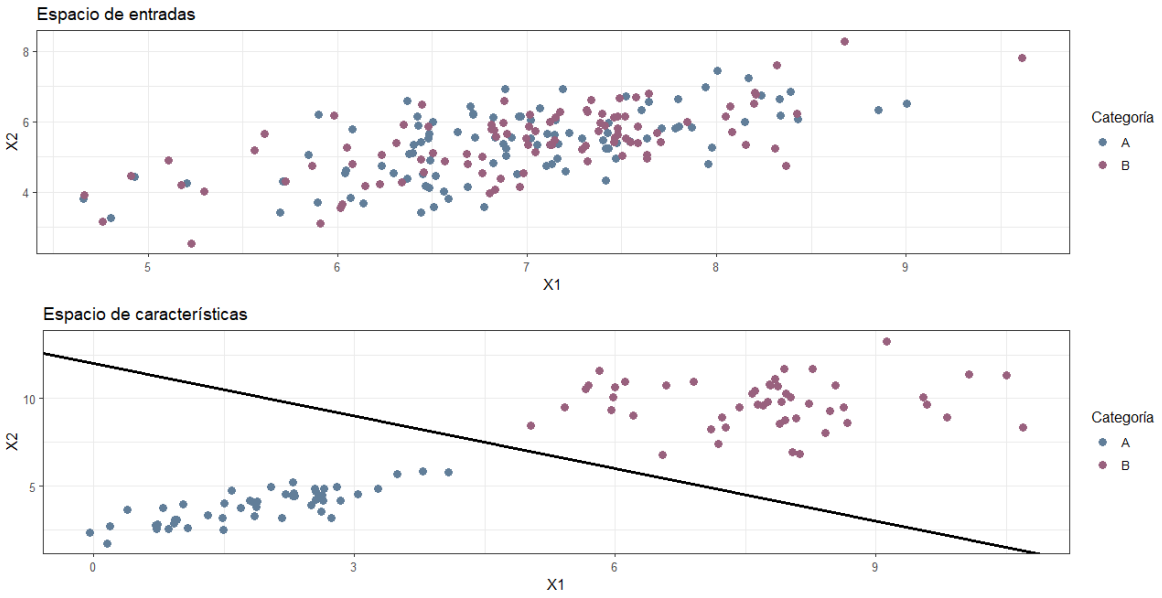

Fig. 3.2: La transformación ϕ nos permite pasar del espacio de entradas a un nuevo espacio de características donde los datos pueden separarse linealmente por un hiperplano.

La teoría de Espacios de Hilbert con Núcleo Reproductor [19] muestra que las funciones Kernel corresponden con un producto escalar y que éste induce un espacio lineal con mayor dimensión que el espacio original, posiblemente infinita. El siguiente teorema muestra cómo transformar el espacio de entradas (de dimensión finita) a otro espacio de dimensión infinita.

Teorema 3 (Teorema de Aronszajn)**.** *Para cualquier función $K : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ que sea simétrica y semidefinida positiva, existe un espacio de Hilbert y una función $\phi : \mathbb{X} \to \mathcal{F}$ tal que*

$$K(x, x') = \langle \phi(x), \phi(x') \rangle \qquad \forall x, x' \in \mathbb{X}$$

Sin embargo, Boser, Guyon y Vapnik demostraron en su artículo *A training algorithm for optimal margin classifiers* [20] que como consecuencia del teorema de Aronszajn, para construir una función kernel no es necesario hacerlo a partir de un conjunto de funciones base $\phi(x) = [\phi_1(x), ..., \phi_m(x)]$, sino que, basta con simplemente definir una función que cumpla las dos condiciones del teorema. De esta forma el kernel representa el producto escalar $\langle \phi(x), \phi(x') \rangle$ que induce un espacio de alta dimensión. Por lo tanto, para evaluar una función kernel no se necesitará conocer dicho conjunto de funciones base.

Este hecho permite reproducir cualquier algoritmo lineal en un espacio de Hilbert o su equivalente no lineal con una transformación no lineal ϕ. Algunas funciones kernel son:

- Kernel lineal: $K(x, x') = \langle x, x' \rangle$
- Kernel polinomial de grado p: $K_p(x, x') = (\gamma \langle x, x' \rangle + \tau)^p$
- Kernel gaussiano: $K(x, x') = \exp\left(-\gamma ||x - x'||^2\right)$

- Kernel sigmoidal: $K(x, x') = \tanh(\gamma\langle x, x'\rangle + \tau)$

Donde γ, τ y p son los parámetros del kernel.

Volviendo a la formulación del problema en el caso no linealmente separable, la idea es construir un hiperplano de separación lineal, ya no en el espacio de entradas, sino en el espacio de las características. La frontera de decisión lineal obtenida en el nuevo espacio se transformará en una frontera de decisión no lineal en el espacio original de entradas. Bajo este contexto el hiperplano separador en el espacio de características está dado por

$$D(x) = w_1\phi_1(x) + ... + w_m\phi_m(x) = \langle w, \phi(x)\rangle$$

donde el parámetro b se ha omitido debido a que puede incluirse en la base de funciones de transformación con la función constante $\phi_1(x) = 1$.

El planteamiento del problema es igual que en las secciones anteriores. Basta con sustituir en el dual (3.16) el producto escalar por la función Kernel:

$$\begin{aligned} \text{máx} \quad & \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i, x_j)\rangle \\ s.a \quad & \sum_{i=1}^{n}\alpha_i y_i = 0, \\ & C \geq \alpha_i \geq 0, \qquad i = 1, ..., n \end{aligned} \tag{3.18}$$

La razón por la que, ahora, el problema de optimización se expresa únicamente en su forma dual, es la posible dimensionalidad infinita del espacio de características, ya que la solución del dual no depende de la dimensionalidad del espacio, sino de la cardinalidad del conjunto de tuplas.

La función de decisión se define como:

$$D(x) = \sum_{i=1}^{n}\alpha_i^* y_i K(x, x_i) \qquad i = 1, ..., n$$

donde el valor de los parámetros α_i se obtendrá como solución al problema de optimización cuadrática dada por (3.18). Cabe destacar que no existe una forma teórica de encontrar el valor de C ni del resto de parámetros del kernel; sin embargo, la forma de determinar el valor de tales parámetros se basa en técnicas de validación cruzada [21].

Capítulo 4

Árboles de decisión

Los árboles de decisión son modelos ampliamente usados para tareas de clasificación y regresión. Se trata de una técnica de aprendizaje supervisado, basada en diagramas de flujo, y que esencialmente aprenden una jerarquía de preguntas «si / si no», lo que lleva a tomar una decisión.

Estos árboles son utilizados para clasificar datos de acuerdo a atributos que pueden ser categóricos o numéricos. Por ejemplo, se puede decidir si un hongo es comestible o no dados distintos atributos físicos; esos atributos pueden ser binarios (si el hongo tiene o no láminas), de múltiples categorías (los colores que presenta), o numéricos (como la altura del hongo).

Se puede explicar el mecanismo de los árboles con preguntas para distinguir un conjunto de animales. Imagina que quieres distinguir entre cuatro animales: osos, halcones, pingüinos y delfines. La meta es llegar a la respuesta correcta preguntando tan pocas preguntas «si/si no» como sea posible [4]. Podrías iniciar preguntando si el animal tiene plumas, una pregunta que reduce a los animales a solo dos posibles grupos. Si la respuesta es «sí», podrías hacer otra pregunta que ayude a distinguir entre halcones y pingüinos. Por ejemplo, podrías preguntar si el animal puede volar. Si el animal no tiene plumas, el animal en cuestión debe ser un delfín o un oso, y entonces requerirás de hacer una pregunta que permita distinguir entre esos dos animales, por ejemplo, preguntar si el animal tiene aletas. El árbol de decisión construido a partir de las preguntas mencionadas se puede visualizar en la Figura 4.1.

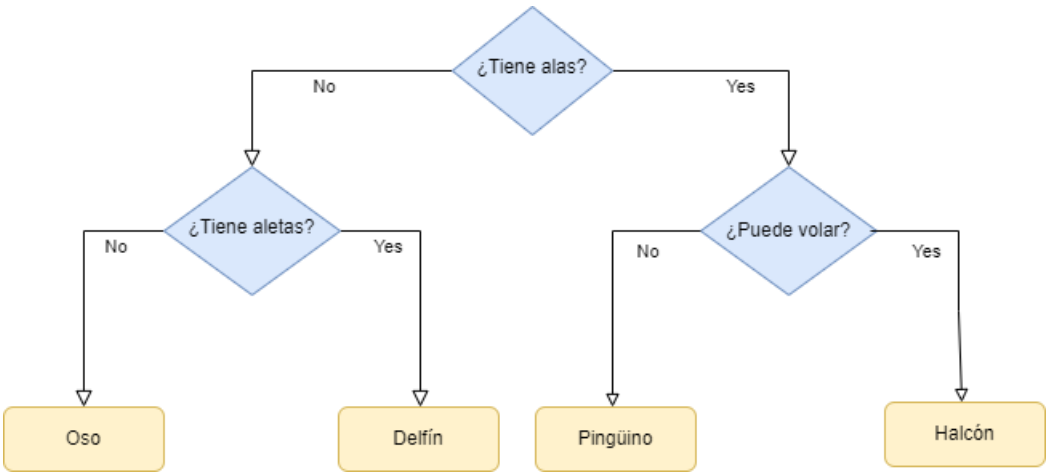

Fig. 4.1: Árbol de decisión para distinguir a los animales mencionados. Cada nodo en el árbol representa una pregunta o un nodo terminal (llamado *hoja*) que contiene la respuesta. Las aristas conectan a una pregunta, con base en la respuesta, a la siguiente pregunta que se podría hacer.

Las ventajas que ofrecen los árboles de decisión son las siguientes:

1. Son fáciles de interpretar y de representar gráficamente, incluso si existen múltiples predictores.
2. Pueden funcionar con tantos predictores numéricos como categóricos se tengan.
3. Al tratarse de métodos no paramétricos, no es necesario cumplir algún tipo de distribución de probabilidad específica.
4. Por lo general, requieren mucha menos limpieza y pre-procesamiento de datos, en comparación con otros métodos de aprendizaje automático [22].
5. No se ven muy influenciados por valores atípicos.
6. Se seleccionan los predictores de manera automática.

Sin embargo, aunque estos modelos presentan múltiples ventajas, no están exentos de desventajas:

1. La capacidad predictiva de un único árbol suele ser inferior en comparación a otros modelos.
2. Los árboles frecuentemente tienden al sobreajuste, por lo que pueden tener un mal rendimiento cuando se les presentan nuevos datos.
3. Son sensibles ante datos de entrenamiento donde una clase es dominante sobre las demás.
4. La construcción de distintos árboles puede cambiar incluso si se utilizan los mismos datos de entrenamiento, y por tanto, también cambia su rendimiento.
5. Cuando se utilizan predictores continuos, se pierde parte de su información al categorizarlos en el momento de la división de los nodos [22].

4.1. Construcción de los árboles

La construcción de un árbol de decisión a partir de un conjunto de entrenamiento se hace siguiendo el *Algoritmo de Hunt*. Dado un conjunto de tuplas[1] de entrenamiento D_t que llega a un nodo t, se sigue el procedimiento:

1. Si D_t contiene registros que pertenecen a una misma clase y_t, entonces t es un nodo hoja etiquetado con y_t.
2. Si D_t es un conjunto vacío, entonces t es un nodo hoja etiquetado con la clase *default* y_d.
3. Si D_t contiene registros pertenecientes a más de una clase, se utiliza un atributo de prueba para dividir los datos en subconjuntos más pequeños.
4. Aplicar el procedimiento de forma recursiva a cada subconjunto.

En otras palabras, el objetivo es generar un árbol de decisión a partir de un conjunto de tuplas de entrenamiento dada una partición D. Los datos de entrada para el algoritmo se componen de la partición de datos D (en esta partición las tuplas de entrenamiento están asociadas a etiquetas de clase) y una lista de atributos de partición candidatos. El algoritmo lleva a cabo un método de selección de atributos para determinar el criterio de partición que mejor divida a las tuplas en clases individuales. Derivado de lo anterior se obtiene el árbol de decisión deseado [23].

Las particiones realizadas por los árboles dependen del tipo de atributo que se tenga. Cuando el atributo es categórico, se realiza una partición con respecto al número de categorías presentes. Si el atributo es continuo, se establece un valor numérico a para efectuar la partición (cuando $X \leq a$ y cuando $X > a$). Para atributos de naturaleza binaria la partición es dicotómica (presencia o ausencia del atributo).

Las medidas de selección de atributos son un conjunto de heurísticas para determinar el criterio de partición que mejor divida a un conjunto de datos D (que contiene etiquetas de clase y tuplas de entrenamiento) en clases individuales. Si se desa dividir a D en particiones más pequeñas de acuerdo a los resultados del criterio de partición, idealmente cada partición debería ser más *pura*, lo que significa que las tuplas que pertenecen a una partición determinada son de la misma clase. Dichas medidas de selección proporcionan un *ranking* por cada atributo descrito en las tuplas de entrenamiento proporcionada. Así, el atributo que tiene mejor puntuación para la medida es el que se elige como atributo de partición para las tuplas dadas.

4.2. Índice Gini

Para los árboles de clasificación y regresión (*Classification And Regression Trees*, en inglés) el índice GINI es una medida utilizada para medir la impureza de una partición de datos o un

[1] Definidas como en el capítulo de Máquinas de soporte vectorial.

conjunto de tuplas de entrenamiento D. El índice se define de la siguiente manera:

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

donde p_i es la probabilidad de que una tupla en D pertenezca a una clase C_i. Dicha probabilidad se estima a partir de efectuar el cociente de el número de tuplas pertenecientes a la clase C_i en la partición D entre el número total de tuplas en la partición D, es decir,

$$p_i = \frac{|C_{i,D}|}{|D|}$$

El índice Gini solo toma en cuenta particiones binarias para cada atributo. Para determinar la mejor partición binaria sobre un atributo A con distintos valores discretos $\{a_1, ..., a_n\}$, es necesario examinar todos los posibles subconjuntos que pueden formarse utilizando los valores conocidos de A:

- Cada subconjunto S_A puede ser considerado como una prueba binaria sobre el atributo A, preguntando si A pertenece a S_A.
- Si A tiene n posibles valores, se tendrían entonces 2^n posibles subconjuntos, lo cual generaría un subconjunto con todos los atributos (el total) y un subconjunto sin atributo alguno (el vacío), los cuales son eliminados debido a que conceptualmente ninguno de ellos representa una partición.
- De esta forma se tienen $2^n - 2$ formas de crear particiones binarias.

Cuando se considera una partición binaria, es preciso calcular una suma ponderada de la impureza de cada partición resultante. Por ejemplo, si una partición binaria sobre A divide a D en D_1 y D_2, el índice GINI de cada partición está dado por

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

El cálculo se hace para cada atributo, y para el caso de valores discretos, el subconjunto que proporcione el menor índice GINI se selecciona como atributo de partición. Para atributos que tienen valores continuos, cada posible punto de partición debe considerarse y se utiliza la misma estrategia que para la ganancia de información [24].

Otro criterio para la selección de atributos para la partición es la *reducción de impureza*. Puede utilizarse para realizar particiones binarias en atributos con valores continuos o discretos. Su expresión está dada por:

$$\Delta Gini(A) = Gini(D) - Gini_A(D)$$

De esta forma, se selecciona como atributo de partición aquel que maximice la reducción de

impureza.

4.3. Sobreajuste, condiciones de paro y poda

Si se tienen muchos atributos es posible que al bajar por los niveles del árbol existan múltiples hojas con pocos registros. El peligro latente es que los resultados podrían no ser representativos para otros datos que no sean de entrenamiento. En otras palabras, se tendría una situación de sobreajuste y todo lo que el árbol está haciendo es memorizar los datos que se le han dado. Para reducir la posibilidad de que esto ocurra es común introducir una condición de paro que establezca con base en un cierto número de registros (un número absoluto o una fracción de todo el conjunto de datos) cuando dejar de dividir a los datos y de crear hojas. De esta manera los resultados pueden seguir siendo estadísticamente significativos. Esto significa que se terminaría con una clasificación confiable en lugar de una no confiable y posiblemente determinista. De esta manera, también se evitaría que el árbol se ajuste bien a los datos de entrenamiento, pero tenga un mal rendimiento con los datos de prueba.

Una condición de paro también puede ayudar cuando se tiene un nuevo dato que se desea clasificar pero no existía ningún dato en el conjunto de entrenamiento con las mismas características.

Otras dos estrategias comunes para prevenir el sobreajuste son: detener la creación del árbol temprano (técnica llamada «poda previa» o «pre-poda»), o construir el árbol y posteriormente remover o colapsar nodos que contengan poca información (técnica llamada «post-poda» o simplemente «poda»). Los posibles criterios para la poda previa incluyen limitar la profundidad máxima del árbol, limitar el número de hojas o exigir un número mínimo de registros en un nodo para seguir dividiéndolo. Si no se restringe la profundidad de un árbol, éste puede volverse arbitrariamente profundo y complejo. Por lo tanto, los árboles sin podar son propensos al sobreajuste y a no generalizar bien a nuevos datos.

La ventaja que ofrece la poda de árboles es que se obtienen árboles más pequeños y menos complejos (por lo cual son más fáciles de interpretar); además, estos árboles suelen ser más rápidos y mejores para hacer la clasificación independientemente de los datos de prueba.

La post-poda se utiliza con mayor frecuencia que la poda previa. Su objetivo es remover sub-árboles de un árbol que ha crecido mucho. El procedimiento de post-poda es el siguiente:

- Se permite que los datos se sobreajusten, y después se poda reemplazando sub-árboles por una hoja. Reemplazar significa retirar todas las ramas y sustituir por un nodo hoja.
- La hoja se etiqueta con la clase que se presenta con mayor frecuencia entre las clases del sub-árbol que fue reemplazado.

- Se poda sólo si el árbol podado resultante mejora o iguala el rendimiento del árbol original sobre el conjunto de prueba.

El proceso es iterativo, escogiendo siempre el nodo a podar que mejore la precisión en el conjunto de prueba hasta el momento en el cual la precisión disminuya.

Aunque los procedimientos de poda producen árboles más compactos, todavía pueden ser bastante grandes y complejos. Por otro lado, un árbol de decisión (independientemente de si ha sido podado o no) puede sufrir de efectos de repetición y duplicación, es decir, dentro del árbol pueden existir sub-árboles exactamente iguales.

4.4. Importancia de los atributos

En lugar de analizar todo el árbol, lo que puede resultar agotador debido a su complejidad y profundidad, existen algunas propiedades útiles que se pueden derivar para resumir el funcionamiento del árbol. El resumen más utilizado es la *importancia de los atributos*, que califica la importancia de cada atributo para la decisión que el árbol toma. La importancia es un número entre cero y uno para cada atributo, donde cero significa que el atributo no se utiliza en absoluto, y uno significa que el atributo predice perfectamente al objetivo [4]. La importancia de los atributos siempre suma uno.

En contraste con los coeficientes de regresión en los modelos lineales, la importancia de los atributos siempre es positiva, y no codifica la clase para la cual el atributo es indicativo. La importancia de los atributos expresa que «la peor proporción» es importante, pero no dice si una proporción alta es indicativa de una muestra benigna o maligna; incluso puede ocurrir que no exista una relación simple entre atributos y clases.

4.5. Bosques aleatorios y agregación Bootstrap

Los *Ensambles* son métodos que combinan varios modelos de aprendizaje automático para crear modelos más potentes. Existen múltiples modelos en la literatura sobre aprendizaje de máquina que pertenencen a dicha categoría, sin embargo, sólo nos centraremos en dos modelos bastante eficaces que se relacionan con los árboles de decisión: bosques aleatorios y agregación bootstrap.

Un bosque aleatorio es esencialmente una colección de árboles de decisión, donde cada árbol es ligeramente distinto a los demás. La idea detrás de los bosques aleatorios es que cada árbol podría hacer relativamente bien su trabajo de predicción, pero probablemente se sobreajustará en parte a los datos. Si se construyen muchos árboles, los cuales funcionan bien y se sobreajustan de distintas maneras, podemos reducir la cantidad de sobreajuste promediando sus resultados. Esta

reducción en el sobreajuste, conservado el poder predictivo de los árboles, se puede demostrar utilizando métodos matemáticos rigurosos.

La agregación bootstrap, *bagging* o empaquetado es un método que consiste en tomar una muestra, con reemplazo, del conjunto de datos completo y luego aplicar la técnica de aprendizaje que se esté utilizando con el fin de obtener un pronóstico, y posteriormente repetir el mismo proceso con otra muestra aleatoria. Este procedimiento se repite varias ocasiones y el pronóstico final será el voto mayoritario en un problema de clasificación, o bien, un promedio en el caso de una regresión.

Capítulo 5

Bondad de ajuste

Las pruebas de bondad de ajuste se definen como pruebas estadísticas para determinar si existe una diferencia significativa entre los datos observados y una distribución de probabilidad (como lo puede ser la distribución normal, poisson, binomial, gamma, entre otras) tomada bajo cierta hipótesis para describir la distribución observada. Ajustar los datos observados a una distribución de probabilidad conocida nos permite atribuir las propiedades de dicha distribución a los datos, haciendo fácil el cálculo de probabilidades, medidas probabilísticas (media, varianza, percentiles), o bien, la estimación de parámetros.

La Bondad de ajuste para los problemas de regresión (donde la variable respuesta es continua) incluye medidas como el error cuadrático medio, el error medio absoluto, coeficientes de correlación y coeficientes de determinación; así como pruebas para ajuste de distribución de parámetros o residuales, pruebas de correlación y pruebas para la varianza del modelo ajustado. Las medidas permiten observar que tan bueno es el ajuste del modelo en cuestión respecto a los datos observados; mientras que las pruebas sirven para verificar que se cumplan los supuestos del modelo.

Para problemas de clasificación (donde la variable respuesta es del tipo categórica) la bondad de ajuste incluye pruebas de significancia (como la prueba Ji-cuadrada de Pearson), medidas que permiten observar el ajuste del modelo (precisión, sensibilidad, especificidad, etc.) y medidas que permiten comparar distintos modelos ajustados a los datos (AIC, devianza, BIC, etc.).

5.1. Devianza

Para un modelo lineal generalizado con observaciones $y = (y_1, ..., y_n)$, sea $L(\mu; y)$ la cual denota la función de log-verosimilitud expresada en términos de las medias $\mu = (\mu_1, ..., \mu_n)$. Sea $L(\hat{\mu}; y)$ la cual indica el máximo de la función de log-verosimilitud para el modelo. Si consideramos todos los posibles modelos, la máxima log-verosimilitud alcanzable es $L(y; y)$. Esto ocurre para la mayoría de los modelos generales, teniendo un parámetro separado para cada observación y el ajuste perfecto $\hat{\mu} = y$.

Este modelo es llamado *modelo saturado*. Dicho modelo explica toda la variabilidad por el predictor lineal del modelo. Aunque un ajuste perfecto es deseable, el modelo saturado no es útil, debido a que no suaviza los datos ni tiene las ventajas que un modelo más simple presenta debido a su parsimonia, como una mejor estimación de una relación verdadera. Sin embargo, frecuentemente el modelo saturado sirve como base para la comparación con otros ajustes del modelo, así como para verificar la bondad de ajuste [25].

Dado lo anterior, la devianza −denotada como $D(y;\mu)$− se define de la siguiente manera:

$$D(y;\mu) = -2[L(\hat{\mu};y) - L(y;y)]$$

Es decir, la devianza se obtiene al multiplicar por -2 el logaritmo del cociente de la máxima verosimilitud del modelo entre la máxima verosimilitud del modelo saturado.

El uso principal de la devianza es para la comparación de los distintos modelos ajustados; entre más grande sea ésta, peor es el ajuste [25].

5.2. AIC

El criterio de información de Akaike (*AIC*, por sus siglas en inglés) juzga a un modelo por qué tan cerca podemos esperar que su muestra se adecue al ajuste real del modelo. Se define como una medida de bondad de ajuste caracterizado por ser un estimador asintótico insesgado respecto a la esperanza de la función de log-verosimilitud [26]. Su expresión generalizada es:

$$AIC = -2[L(f(x;\theta)) - P]$$

donde P es el número de parámetros en el modelo y $L(f(x;\theta))$ denota a la log-verosimilitud de las variables explicativas del modelo.

Aunque la función de restar el número de parámetros en el modelo es para ajustar el sesgo, el AIC esencialmente penaliza un modelo por tener muchos parámetros. Entre menor sea el AIC, mejor es el ajuste del modelo [25].

5.3. Matriz de confusión

Para modelos donde se desea clasificar de forma binaria a los datos, se define a la *matriz de confusión*[1] como una matriz de 2×2 que contrasta los datos observados contra los predichos por el modelo. En la diagonal de la matriz encontramos los casos bien clasificados, mientras que en las dos celdas restantes encontramos los casos mal clasificados.

[1] También llamada matriz de clasificación o matriz de predicción.

Supongamos que el modelo clasifica a los datos en dos categorías, donde la principal categoría de interés es la categoría E. Para fines prácticos, cada celda de la matriz de confusión tiene un nombre debido a la clasificación y a la predicción realizada por el modelo (véase la Tabla 5.1):

- Los verdaderos positivos (VP) son los casos que pertenecen a la categoría E y el modelo clasificó correctamente en dicha categoría.
- Los falsos negativos (FN) son los casos que pertenecen a la categoría E pero el modelo no clasificó en tal categoría.
- Los verdaderos negativos (VN) son los casos que no pertenecen a la categoría E y el modelo no clasificó en esa categoría.
- Los falsos positivos (FP) son los casos que no pertenecen a la categoría E; sin embargo, el modelo sí los clasificó en dicha categoría.

	Predichos	
Observados	Positivo	Negativo
Positivo	VP	FN
Negativo	FP	VN

Tabla 5.1: Matriz de confusión para una clasificación binaria.

5.4. Precisión

La precisión (*Accuracy* en inglés) se define como el cociente de el número de datos clasificados correctamente entre el número total de observaciones. La precisión indica la proporción de datos clasificados de manera correcta por el modelo. En términos de la matriz de confusión, la precisión se calcula como:

$$\text{Precisión} = \frac{VP + VN}{VP + VN + FP + FN}$$

La precisión toma valores entre cero y uno. Si la precisión tiende a valores cercanos a uno, el modelo clasifica de manera óptima a los datos; por el contrario, si la precisión tiende a cero, el modelo no clasifica bien a los datos.

5.5. Sensibilidad

La sensibilidad (*sensitivity* en inglés) indica la capacidad del modelo para clasificar correctamente a los casos positivos como casos que efectivamente lo son. En otras palabras, la

sensibilidad mide la proporción de verdaderos positivos que son clasificados correctamente como positivos. En términos de la matriz de confusión, la sensibiliad se calcula de la siguiente manera:

$$Sensibilidad = \frac{VP}{VP + FN}$$

La sensibilidad también toma valores entre cero y uno. Entre mayor sea la sensibilidad, el modelo clasifica de mejor manera a los casos positivos en dicha categoría.

5.6. Especificidad

La especificidad (*specificity* en inglés) indica la capacidad del modelo para clasificar como casos negativos a los casos que efectivamente son negativos. Es decir, la especificidad mide la proporción de verdaderos negativos que son identificados correctamente como negativos. En términos de la matriz de confusión, la especificidad se calcula de la siguiente forma:

$$Especificidad = \frac{VN}{VN + FP}$$

Al igual que las medidas anteriores, la especificidad está acotada entre cero y uno. Entre mayor sea la especificidad, el modelo clasifica de mejor forma a los casos negativos en tal categoría.

5.7. Curva ROC y AUC

La curva característica operativa del receptor (curva *ROC*, por sus siglas en inglés) representa la especificidad frente a la sensibilidad para cada posible punto de corte en la escala de resultados del modelo en estudio. Para graficar la curva ROC, se representa a $1 - especificidad$ en el eje X y a la *sensibilidad* en el eje Y. Puesto que podemos ver a la especificidad y a la sensibilidad como probabilidades, la curva ROC está contenida en el cuadrado $[0, 1] \times [0, 1]$.

Cuando las proporciones son iguales entonces no se obtiene una curva, sino una diagonal, la cual indica que el modelo no hace una correcta distinción de los casos verdaderos positivos ni de los casos verdaderos negativos. Si la curva ROC obtenida se aproxima a la esquina superior izquierda del cuadrado unitario, el poder predictivo del modelo es bueno (existe una gran cantidad de verdaderos positivos y verdaderos negativos); por otro lado, si la curva se aproxima a la diagonal, el modelo tiene un pobre poder predictivo (existe una gran cantidad de falsos positivos y falsos negativos).

Por consecuencia de la curva ROC, surge el *Área bajo la curva* (AUC, por sus siglas en inglés), una métrica que nos permite medir la capacidad discriminante de la prueba. El rango de valores de AUC va desde 0.5, siendo este valor un indicador de que el modelo carece de una capacidad discriminante, hasta uno, que se obtiene cuando las dos categorías están perfectamente diferenciadas por el modelo. De manera general, se dice que un AUC que toma valores en el

intervalo $[0.5, 0.7)$ tiene baja exactitud, un AUC en el intervalo $[0.7, 0.9)$ puede ser útil para ciertos propósitos, y un AUC mayor a 0.9 tiene una alta exactitud [27].

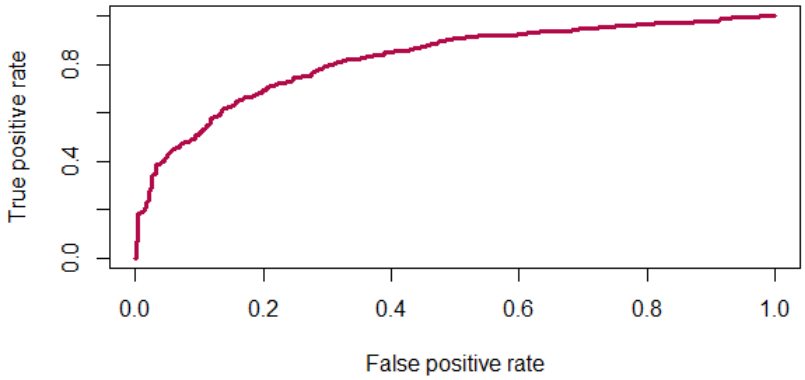

Fig. 5.1: Ejemplo de Curva ROC.

Capítulo 6

Correlación y multicolinealidad

6.1. Covarianza y correlación

La covarianza es una medida de dependencia entre dos variables aleatorias. Dadas X, Y variables aleatorias, la covarianza (teórica) es definina por

$$\sigma_{XY} = Cov(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])] \tag{6.1}$$

Por propiedades de la esperanza y un poco de álgebra, la ecuación (6.1) puede escribirse como

$$\sigma_{XY} = Cov(X, Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y] \tag{6.2}$$

Si X, Y son independientes, entonces la covarianza entre ellas es igual a cero[1]; no obstante, la afirmación inversa no necesariamente es verdadera. La covarianza de X consigo misma es la varianza, es decir,

$$\sigma_{XX} = Cov(X, X) = Var(X)$$

Si X es un vector aleatorio p-dimensional, $X = (X_1, ..., X_p)^T$, entonces la covarianza (teórica) entre todos los elementos se define de forma matricial, en una matriz llamada *matriz de covianza*:

$$\Sigma = \begin{pmatrix} \sigma_{X_1X_1} & \cdots & \sigma_{X_1X_p} \\ \vdots & \ddots & \vdots \\ \sigma_{X_pX_1} & \cdots & \sigma_{X_pX_p} \end{pmatrix}$$

Cuando se tienen muestras aleatorias, se recurre a la definición de *varianza muestral* y *covarianza muestral*, las cuales se definen, respectivamente, como

$$S_{XX} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2 \tag{6.3}$$

[1]Dicha proposición puede demostrarse utilizando la ecuación (6.2).

$$S_{XY} = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y}) \tag{6.4}$$

Frecuentemente se suele reemplazar al factor $\frac{1}{n}$ en las ecuaciones (6.3) y (6.4) por $\frac{1}{n-1}$ para lograr un sesgo pequeño o bien, para que sea nulo; por ejemplo, S_{XX} con el factor $\frac{1}{n-1}$ resulta ser un estimador insesgado[2] para la varianza de una distribución normal.

Para un vector aleatorio p-dimensional, se define la *matriz de covarianza muestral* como:

$$S = \begin{pmatrix} S_{X_1X_1} & \cdots & S_{X_1X_p} \\ \vdots & \ddots & \vdots \\ S_{X_pX_1} & \cdots & S_{X_pX_p} \end{pmatrix}$$

Para un diagrama de dispersión de dos variables, la covarianza mide «qué tan cerca está la dispersión de una línea». En este sentido la covarianza mide solo la «dependencia lineal». Una covarianza positiva corresponde a un diagrama de dispersión con pendiente ascendente; una covarianza negativa corresponde a un diagrama de dispersión con pendiente descendiente[28] (véase Figura 6.1).

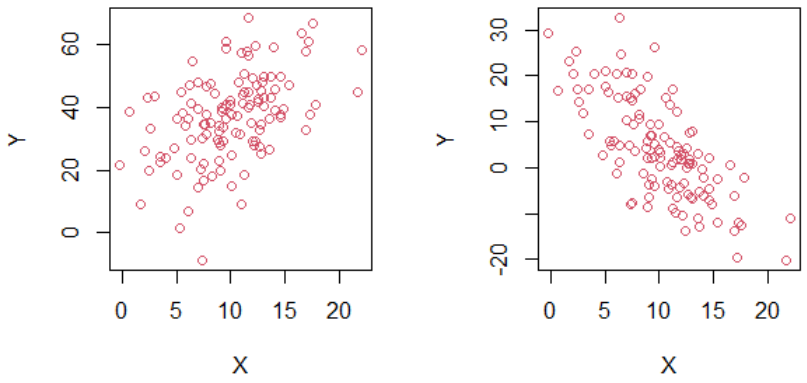

Fig. 6.1: Diagramas de dispersión con covarianza positiva (izquierda) y covarianza negativa (derecha).

La correlación entre dos variables aleatorias X,Y es definida en términos de la covarianza y las varianzas de las variables, como se ve a continuación:

$$\rho_{XY} = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}$$

[2]Se dice que un estimador $\hat{\theta}$ es insesgado si el valor esperado del estimador es igual al parámetro verdadero, es decir, $\mathbb{E}[\hat{\theta}] = \theta$.

La ventaja de la correlación frente a la covarianza es que la correlación es independiente de la escala, es decir, cambiar la escala de medición de las variables no cambia el valor de la correlación. Por lo tanto, la correlación es más útil que la covarianza como medida de asociación entre dos variables aleatorias [28].

La versión muestral de la correlación se define como:

$$r_{XY} = \frac{S_{XY}}{\sqrt{S_{XX}S_{YY}}}$$

La correlación en valor absoluto siempre es menor o igual a 1. La igualdad a 1 se logra cuando se calcula la correlación de una variable consigo misma. Por su parte, la covarianza es igual a cero si la correlación es igual a cero y viceversa.

Para un vector p-dimensional, $X = (X_1, ..., X_p)'$, se define a la *matriz de correlación*

$$\mathcal{P} = \begin{pmatrix} \rho_{X_1X_1} & \cdots & \rho_{X_1X_p} \\ \vdots & \ddots & \vdots \\ \rho_{X_pX_1} & \cdots & \rho_{X_pX_p} \end{pmatrix}$$

y se define la versión muestral de la matriz de correlación

$$\mathcal{R} = \begin{pmatrix} r_{X_1X_1} & \cdots & r_{X_1X_p} \\ \vdots & \ddots & \vdots \\ r_{X_pX_1} & \cdots & r_{X_pX_p} \end{pmatrix}$$

6.2. Multicolinealidad

La multicolinealidad es un problema muestral que consiste en la existencia de una relación lineal entre dos o más covariables explicativas X_i. Dependiendo de cómo sea dicha relación lineal se hablará de multicolinealidad perfecta o aproximada. Las principales causas que producen multicolinealidad en un modelo son:

- Relación causal entre variables explicativas del modelo.
- Escasa variabilidad en las observaciones de las variables explicativas.
- Reducido tamaño de la muestra.

La multicolinealidad exacta o perfecta hace referencia a la existencia de una relación lineal exacta entre dos o más variables independientes. La multicolinealidad aproximada hace referencia a la existencia de una relación lineal aproximada entre dos o más variables independientes [29]. En consecuencia, cuando existen problemas de multicolinealidad se presentan las siguientes cuestiones:

- Las varianzas de los estimadores son muy grandes.
- Los coeficientes estimados serán muy sensibles ante pequeños cambios en los datos.
- Las covarianzas y correlaciones entre las variables explicativas son fuertes.
- Los intervalos de confianza tienden a ser muy amplios.
- Los signos de los coeficientes estimados pueden ser contraintuitivos.

6.3. Detección y solución al problema de multicolinealidad

La detección de la multicolinealidad puede hacerse con los siguientes métodos:

- Examinación de la matriz de correlación. En general, una alta correlación entre dos variables explicativas puede dar evidencia de un problema de multicolinealidad; sin embargo, la ausencia de correlaciones altas no es evidencia de que no haya problemas de multicolinealidad. Puede existir baja correlación entre dos de ellas, pero alta correlación tres a tres, cuatro a cuatro, etcétera [30].
- Análisis de valores y vectores propios de $X'X$. Los valores propios de $X'X$, $\lambda_1, ..., \lambda_k$, pueden usarse para medir la extensión de la multicolinealidad en los datos. Si existe una o más dependencias linealmente cercanas en los datos, entonces uno o más de los valores propios serán pequeños. Con frecuencia también es examinado el número de condición (*condition number* en inglés) de $X'X$, el cual se define como

$$\kappa = \frac{\lambda_{\text{máx}}}{\lambda_{\text{mín}}}$$

  Si κ es menor que 100, no existen problemas fuertes de multicolinealidad. Si κ se encuentra entre 100 y 1000, implica una multicolinealidad moderada; y si es mayor que 1000 entonces existen problemas severos de multicolinealidad [30].
- Factor de inflación de la varianza (VIF). El factor de inflación de la varianza se define como

$$VIF_j = C_{jj} = \frac{1}{(1 - R_j^2)}$$

  donde $C = (X'X)^{-1}$ y R_j^2 es el coeficiente de determinación que se obtiene al hacer regresión entre X_j (como variable respuesta) y el resto de los predictores. Si X_j es «independiente» del resto de los predictores, R_j^2 será pequeña y entonces C_{jj} será cercana a uno. En caso contrario, si R_j^2 es cercana a uno, entonces la variable X_j puede ser explicada por los predictores restantes, y así C_{jj} será grande [30].

  Si el VIF es igual o mayor a uno, esto indica que existe multicolinealidad. Generalmente se dice que si cualquiera de los VIF de las variables excede a 5, los coeficientes de regresión asociados no están bien estimados debido a la multicolinealidad.

Algunas de las posibles estrategias para solucionar el problema de multicolinealidad son las siguientes:

- Mejorar el diseño muestral extrayendo la información máxima de las variables observadas.
- Eliminar las variables que se sospechan causantes de la multicolinealidad.
- En caso de disponer de pocas observaciones, aumentar el tamaño de la muestral.
- Re-especificar el modelo. Si algunas covariables explicativas están fuertemente correlacionadas, introducir al modelo una nueva variable en función de ellas.
- Utilizar el método de componentes principales.
- Utilizar la Regresión LASSO para una pre-selección de variables explicativas en el modelo.

Capítulo 7

Análisis estadístico del conjunto de datos `redwine.csv`

7.1. Descripción del conjunto de datos

Para el desarrollo de esta tesis se tomó el conjunto de datos `redwine.csv`[1], el cual consta de 12 variables, 1599 instancias y no hay presencia de valores ausentes. Todas las variables explicativas son del tipo continuas. En caso de la variable respuesta, `Good.quality`, fue propuesta como una variable binaria, la cual tomó como base a la variable original `quality`, dicha variable es del tipo categórica ordinal con valores entre 0 y 10, y que representa el puntaje obtenido por el vino. Se definió a `Good.quality` como 1 si el puntaje del vino (tomado de la variable *quality*) fuese mayor o igual a 6, y como 0 si el puntaje del vino fuese menor o igual a 5.

Las variables explicativas presentes en el conjunto de datos son:

1. **Acidez fija.** Es la acidez presente en los vinos debido a los ácidos orgánicos de la uva: el tartárico, el málico y el cítrico [31].
2. **Acidez volátil.** Acidez originada durante la vinificación, donde se forman cantidades limitadas de ácido acético, y en la fermentación maloláctica que transforma el ácido málico en ácido láctico, mejorando la sensación gustativa [31].
3. **Ácido cítrico.** Ácido orgánico que se encuentra en casi todos los tejidos animales y vegetales. Se presenta en forma de ácido de frutas en el limón, mandarina, lima, toronja, piña, naranja, ciruela, uva; así como en los huesos, músculos y sangre de animales [32].
4. **Azúcar residual.** Cantidad total de azúcar que queda en el vino después la fermentación. Es medida en gramos por litro [31].
5. **Cloruros.** Cantidad total de cloruro de sodio presente en el vino, expresada en gramos por litro [33].

[1] Tomado del sitio https://archive.ics.uci.edu/ml/datasets/Wine+Quality.

6. **Dióxido de azufre libre.** Forma activa de los sulfitos encontrados en vinos. Esta forma será activa como agente antimicrobiano y antioxidante. Se llama libre porque no está ligado a ningún otro compuesto. Se mide en miligramos por litro [34].

7. **Dióxido de azufre total.** Es la suma del dióxido de azufre libre y el dióxido de azufre combinado[2] presentes en el vino. Medido en miligramos por litro [34].

8. **Densidad.** Magnitud que refiere a la cantidad de masa que existe en un determinado volumen. Las unidades más utilizadas son gramos por centímetro cúbico o kilogramo por metro cúbico [31].

9. **pH**. Cantidad que indica la concentración de iones Hidrógeno presentes en una sustancia [31]. El pH se utiliza para clasificar a las sustancias como ácidas si su pH es menor a 7, neutras si el pH es exactamente igual a 7, y alcalinas con un pH mayor a 7.

10. **Sulfitos.** Variantes del dióxido de azufre, que se generan de forma natural en el proceso de fermentación de las levaduras del vino. Los sulfitos tienen funciones conservantes, son antioxidantes, agentes antimicrobianos y antioxidásicos. Son medidos en miligramos por litro [34].

11. **Alcohol.** Alcohol etílico o etanol producto de la fermentación alcohólica, proceso realizado por levaduras pertenecientes al género *Saccharomyces*. El alcohol en las bebidas se registra en el *grado alcohólico*, el cual se define como la proporción de alcohol presente por volumen. La graduación de los vinos varía entre el 7 y 16 % [31].

Para tener un resumen de las variables explicativas respecto a sus valores, se obtuvieron algunas estadísticas descriptivas (mínimo, media, mediana, máximo, desviación estándar y coeficiente de variación[3]) de cada una de ellas. Dicha información puede verse a continuación en la Tabla 7.1.

Estadísticas	A. fija	A. volátil	Á. cítrico	Azúcar res.	Cloruros	SO_2 libre
Mínimo	4.60	0.12	0.00	0.90	0.012	1.00
Mediana	7.90	0.52	0.26	2.20	0.079	14.00
Media	8.32	0.5278	0.271	2.539	0.08747	15.87
Máximo	15.90	1.58	1.00	15.50	0.611	72.00
Desv. est.	1.7410	0.1790	0.1948	1.4099	0.047	10.46
CV	20.9275	33.9243	71.888	55.5350	53.8094	65.8910
Estadísticas	SO_2 total	Densidad	pH	Sulfitos	Alcohol	
Mínimo	6.00	0.9901	2.740	0.330	8.40	
Mediana	38.00	0.9968	3.310	0.620	10.20	
Media	46.47	0.9967	3.311	0.6581	10.42	
Máximo	289.00	1.0037	4.010	2.00	14.90	
Desv. est.	32.8953	0.0018	0.1543	0.1695	1.06566	
CV	70.7916	0.1893	4.6626	25.7551	10.2242	

Tabla 7.1: Estadísticas descriptivas de las variables explicativas.

[2]El dióxido de azufre combinado es aquel que se liga a los azúcares, aldehídos y cetonas presentes en el vino después de agregarse a éste. Es una forma no activa.

[3]El coeficiente de variación fue calculado de la forma $CV = \frac{\sigma}{\bar{X}} * 100$ para interpretarse de forma porcentual.

La Tabla 7.1 indica que en todas las variables explicativas el valor máximo se encuentra considerablemente alejado de la media y la mediana, por lo cual, en primera impresión, se espera la existencia de valores atípicos. Por su parte, con respecto al coeficiente de variación y a la desviación estándar, las variables con mayor dispersión son: Ácido cítrico, Azúcar residual, cloruros, SO_2 libre y SO_2 total. Para la variable respuesta, se registraron 855 vinos de buena calidad y 744 de mala calidad.

7.2. Distribución de las variables

Para visualizar la distribución de las variables explicativas, se realizó el histograma de cada una de ellas (Figura 7.1). Se puede notar que Densidad y pH parecieran tener una asimetría nula, mientras que todas las demás variables presentan una asimetría positiva. Los histogramas también indican colas superiores muy largas para todas las variables, por consiguiente, esperamos la presencia de datos atípicos en la parte superior de todas las variables; asimismo esperamos datos atípicos en la parte inferior de la Densidad y el pH. Finalmente, todas las variables son del tipo unimodal, excepto Acidez volátil y Ácido cítrico, las cuales son bimodal y multimodal respectivamente.

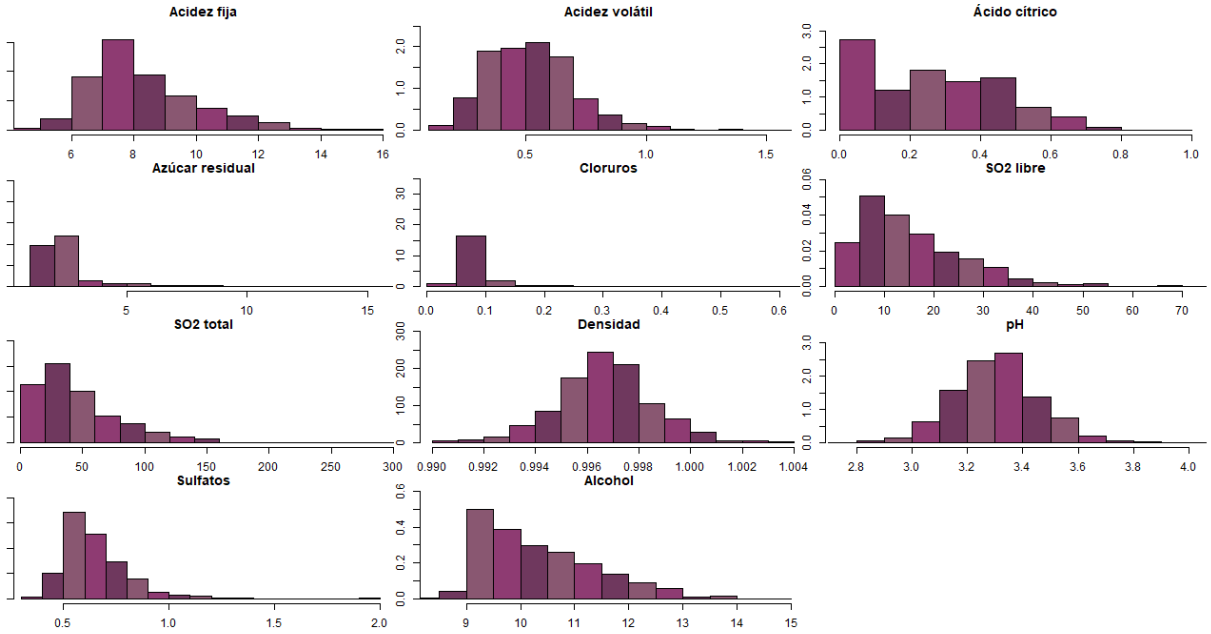

Fig. 7.1: Histogramas de las variables explicativas.

7.3. Datos atípicos y agrupamiento

Debido a que se desea ajustar dos métodos de clasificación binaria, resulta conveniente visualizar si los datos por sí mismos realizan un agrupamiento en dos (o incluso más) categorías.

Para observar si existe un agrupamiento, se realizó la Curva de Andrews[4] (Figura 7.2), en la cual se refleja que los datos no se dividen en dos o más categorías, es decir, por sí mismos los datos no se dividen en grupos, lo que indica que no hay diferencias evidentes entre vinos de buena calidad y de mala calidad. No obstante, se puede observar algunas curvas individuales que se alejan de las demás; esto implica la existencia de valores atípicos extremos.

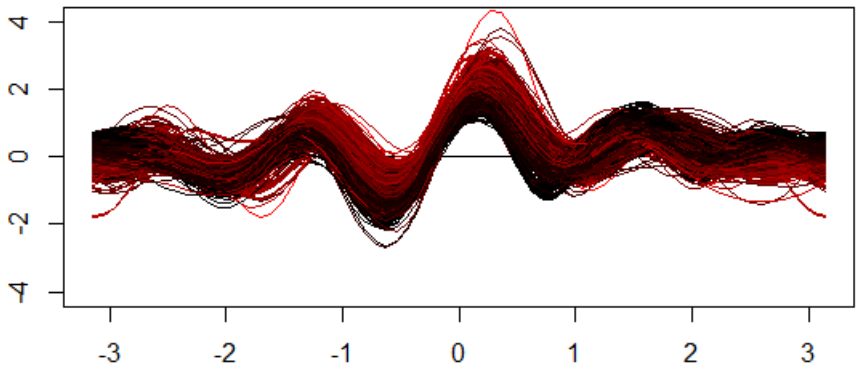

Fig. 7.2: Curva de Andrews para los datos observados.

Para visualizar los datos atípicos, de los cuales ya se había sospechado en las dos secciones anteriores, se llevó a cabo el boxplot para cada una de las variables (véase Figura 7.3). Se puede ver que todas las variables presentan valores atípicos en la cola superior de su distribución, donde Azúcar residual y Cloruros tienen un número bastante grande de valores atípicos, y además, valores sumamente alejados de los máximos. Sulfitos, Acidez volátil, Ácido cítrico, Alcohol, SO_2 libre y SO_2 total muestran algunos valores atípicos muy alejados del resto. Debido a estos comportamientos, es probable que los valores atípicos alteren los ajustes de los modelos, principalmente el ajuste de regresión logística.

[4]Las curvas de Andrews son utilizadas para representar un dato de un espacio multidimensional en una curva de dos dimensiones, empleando una función basada en series de senos y cosenos, donde los términos constantes son definidos por los valores cuantificados para las p variables. Estas curvas permiten hacer una comparación visual de grupos homogéneos a partir de la diferencia entre curvas [35].

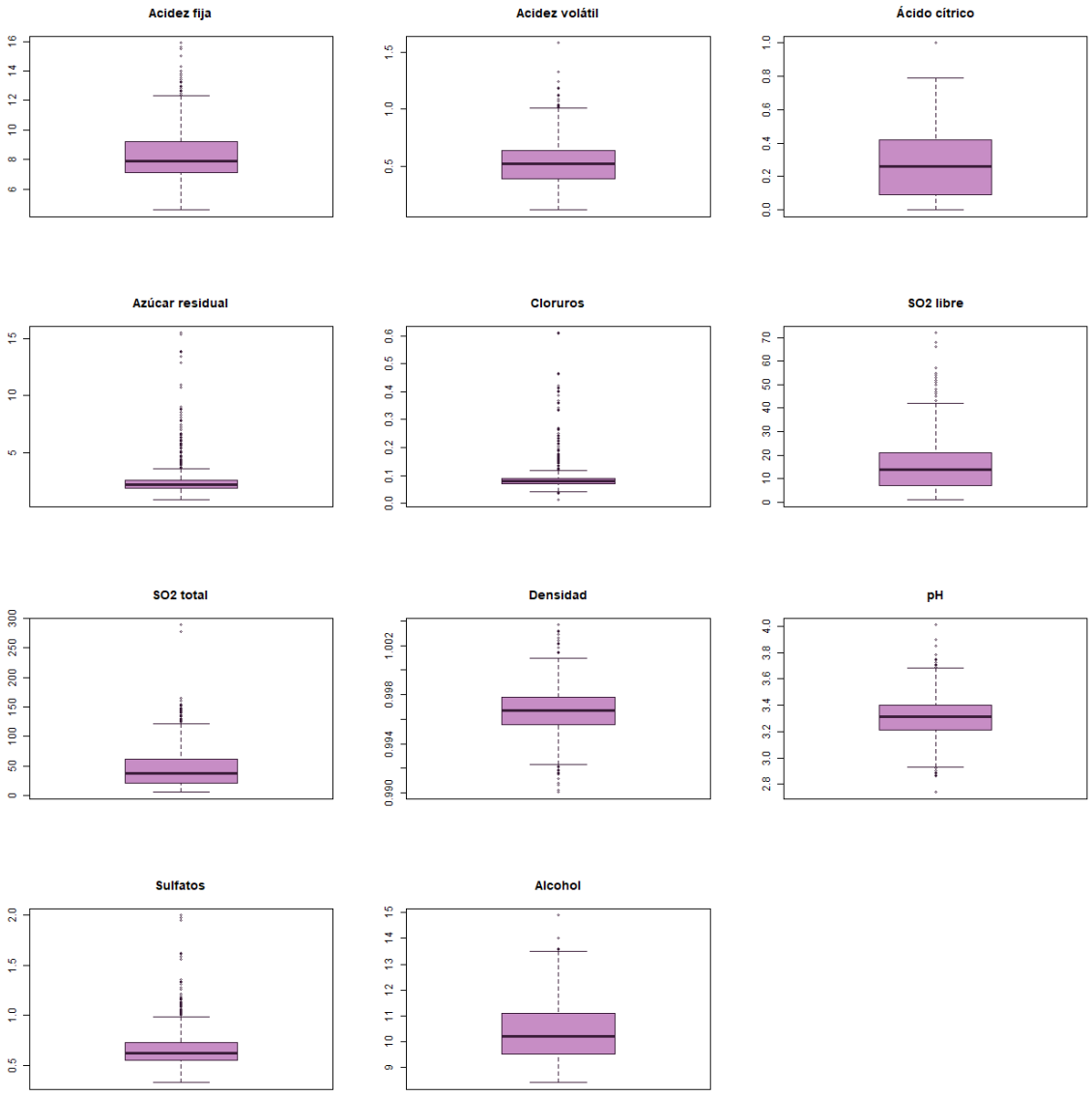

Fig. 7.3: Boxplots de las variables explicativas.

7.4. Correlación

Como se mencionó en el capítulo 6, la multicolinealidad puede ocasionar distintos problemas al momento de ajustar los modelos de regresión. Para verificar la correlación entre las variables explicativas se realizó un diagrama de correlación basado en la matriz de correlación muestral de los datos (Figura 7.4). Podemos ver que existe una fuerte correlación positiva entre Ácido cítrico y Acidez fija (esto ocurre puesto que la acidez fija tiene como componente al ácido cítrico), Acidez fija y Densidad, y Dióxido de azufre libre y Dióxido de azufre total (debido a que el dióxido de azufre total se compone del dióxido libre y combinado); y existe una fuerte correlación negativa

entre Acidez fija y pH (ya que por definición de pH, entre más ácida sea una sustancia, menor es su pH), Ácido volátil y Ácido cítrico, pH y Ácido cítrico, y Densidad y Alcohol. Por lo anterior, se espera que existan problemas de multicolinealidad al ajustar los modelos y será necesario eliminar variables explicativas para tener mejores ajustes.

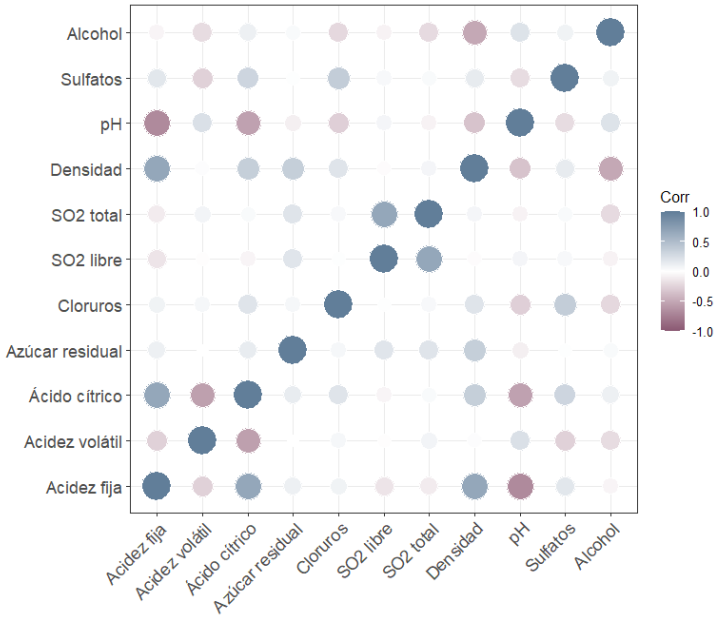

Fig. 7.4: Diagrama de la matriz de correlación muestral de los datos.

7.5. Comparación por grupos

Para concluir el análisis estadístico de los datos, se hizo una comparativa de las medias y medianas entre los vinos de buena calidad y los de mala calidad (véase Tabla 7.3). Esto se realizó con el fin de verificar si existen diferencias significativas entre ambos grupos y para observar, en promedio, los valores que tomaría un buen vino. En promedio, un buen vino tiene mayores índices de acidez fija, ácido cítrico, sulfitos y alcohol; mientras tiene menores índices en acidez volátil, azúcar residual, cloruros, dióxido de azufre libre y total, densidad y pH. Las afirmaciones anteriores también se cumplen si tomamos la mediana para comparar.

Para verificar si existen diferencias significativas entre los grupos, se propuso realizar una *Prueba t* para la comparación de medias. Sin embargo, se encontró que ninguna variable cumplía el supuesto de normalidad[5] (supuesto fundamental para realizar la prueba). Por lo tanto, se recurrió a la *Prueba de Wilcoxon-Mann-Whitney*[6] para así comparar las medias de las categorías. Con un nivel de significancia $\alpha = 0.05$, y con el contraste de hipótesis

[5]Supuesto verificado con la *Prueba de Anderson - Darling* a un nivel de significancia $\alpha = 0.05$. Todos los p-values obtenidos fueron menores a 2.2×10^{-16}.

[6]Prueba no paramétrica para la comparación de medias que no requiere normalidad ni homocedasticidad para ser realizada.

$$H_0 : \mu_1 = \mu_2 \text{ vs } H_1 : \mu_1 \neq \mu_2$$

se obtuvo la siguiente tabla donde se pueden observar las variables con sus respectivos p-values:

Variable	p-value
Acidez fija	0.001193
Acidez volátil	0
Ácido cítrico	0
Azúcar residual	0.5797
Cloruros	0
SO_2 libre	0.03264
SO_2 total	0
Densidad	0
pH	0.8363
Sulfatos	0
Alcohol	0

Tabla 7.2: P-values obtenidos para la comparación de las medias respecto a las variables explicativas.

Se encontró que únicamente las variables Azúcar residual y pH tienen medias estadísticamente iguales. Por consiguiente, se puede concluir que existen diferencias estadísticamente significativas entre los vinos de buena calidad y de mala calidad. En contraste con la homogeneidad de los datos vista con la curva de Andrews, podemos esperar que el rendimiento de los clasificadores no sea tan pobre, pero tampoco la óptima; en otras palabras, se esperaría que la precisión sea, al menos, mayor que 0.5, mas no muy cercana a uno.

Estadísticas	A. fija	A. volátil	Á. cítrico	Azúcar res.	Cloruros	SO_2 libre
Media (MC)	8.1422	0.5895	0.2377	2.5420	0.09298	16.5672
Media (BC)	8.4740	0.4741	0.2998	2.5359	0.08266	15.2725
Mediana (MC)	7.8	0.59	0.22	2.2	0.081	14
Mediana (BC)	8.0	0.46	0.31	2.2	0.077	13
Estadísticas	SO_2 total	Densidad	pH	Sulfitos	Alcohol	
Media (MC)	54.6451	0.9970	3.3116	0.61853	9.9264	
Media (BC)	39.35205	0.9964	3.3106	0.6926	10.8550	
Mediana (MC)	45	0.9969	3.31	0.58	9.7	
Mediana (BC)	33	0.9964	3.31	0.66	10.8	

Tabla 7.3: Comparación de medias y medianas entre los vinos de buena calidad (BC) y mala calidad (MC).

Capítulo 8

Ajuste de los métodos de clasificación

El código utilizado para los distintos ajustes fue implementado en **R**. Para llevar a cabo los ajustes de los métodos de clasificación se hizo una partición del conjunto de datos completo, donde dos tercios se utilizaron para el conjunto de entrenamiento de los modelos y el tercio restante para un conjunto de prueba.

8.1. Ajuste del modelo de regresión logística

Se ajustó un primer modelo de regresión logística utilizando la función `glm()`, donde se especificó que la variable respuesta `Good.quality` se explicaría por todas las demás, el conjunto de datos para el ajuste sería el conjunto de entrenamiento, y la familia del modelo es binomial (debido a que se utilizó regresión logística). Posteriormente se ejecutaron las funciones `summary()`, `confint()`, `vif()` y `step()` para obtener un resumen del modelo, los intervalos de confianza de los coeficientes ajustados, observar la presencia de multicolinealidad y elegir los predictores que deberían removerse para nuevos modelos. Dicho análisis informó que las variables Acidez fija, Azúcar residual, Densidad, pH y Ácido cítrico son estadísticamente no significativas para el modelo, debido a que registraron *p-values* mayores al nivel de significancia e intervalos de confianza que contenían al cero. Por otro lado, Acidez fija y Densidad mostraron factores de inflación de la varianza iguales a 7.954 y 5.690 respectivamente, indicando que existían problemas de multicolinealidad entre los predictores.

Se llevó a cabo un segundo modelo sin considerar a las variables que resultaron estadísticamente no significativas para el modelo anterior. Asimismo, se ejecutaron las funciones utilizadas para en análisis del modelo anterior. El resumen señaló que todos los predictores fueron estadísticamente significativos, con intervalos de confianza que no contuvieron al cero. Finalmente, se encontró que todos los predictores tuvieron factores de inflación de la varianza menores a dos, por lo cual no existieron problemas debido a multicolinealidad.

Para remover los valores atípicos presentes en cada una de las variables explicativas se utilizó el *Criterio de rechazo de Chauvenet*. El criterio consiste en rechazar todas aquellas mediciones cuya probabilidad de aparición sea inferior a $\alpha = \frac{1}{2n}$, con n el tamaño de la muestra.

Esto supone que se deben rechazar aquellos datos cuya desviación a la media sea superior a un determinado valor (función de la desviación estándar muestral) [36]. Por lo tanto, el criterio se simplifica con la siguiente expresión:

$$|X_i - \overline{X}| > K_n S$$

Dado lo anterior, para cada variable explicativa, se eliminaron los datos x_i que no estuvieran en el intervalo $(\overline{X} - K_n S, \overline{X} + K_n S)$, donde K_n es el cuantil $1/4n$ de una distribución normal estándar. En este caso, debido a que $n = 1599$, entonces $K_{1599} = 3.604549$.

Al aplicar el criterio de Chauvenet, se eliminaron 79 datos del conjunto total (lo que representa el 4.94 % del conjunto de datos). Se realizaron nuevamente los histogramas de las variables (Figura 8.1) para observar si se generaron cambios sustanciales en los datos. Se puede observar que, aunque las distribuciones siguen mostrando colas superiores pesadas, no hay presencia de valores atípicos tan extremos como en un principio.

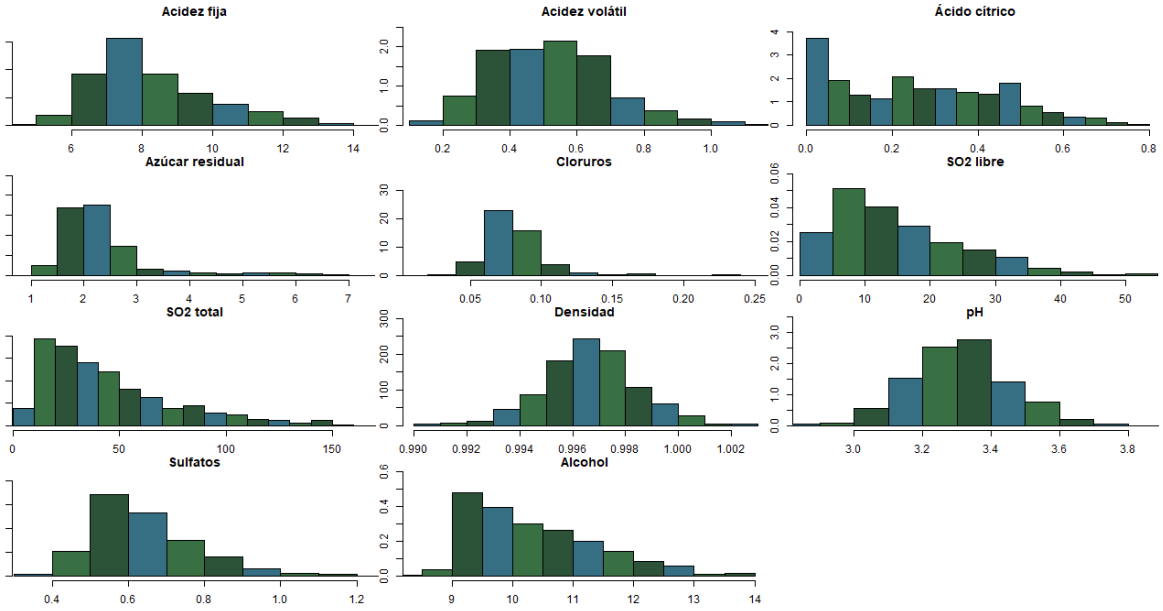

Fig. 8.1: Histogramas de las variables explicativas después de remover valores atípicos.

Con respecto al nuevo conjunto de datos sin los valores atípicos señalados, a éste se le aplicó, como al conjunto total, una partición de dos tercios para el conjunto de entrenamiento y un tercio para el conjunto de prueba.

Se ajustó un tercer modelo logístico siguiendo los pasos de los modelos anteriores, tomando en cuenta al conjunto de entrenamiento con valores removidos. El resumen de este modelo indicó que las variables Acidez fija, Ácido cítrico, Azúcar residual, Cloruros, Dióxido de azufre libre, Densidad, pH no fueron estadísticamente significativas para el modelo. Nuevamente, Acidez fija y

Densidad mostraron factores de inflación de la varianza mayores a 5, denotando problemas de multicolinealidad.

Finalmente, se ejecutó un cuarto modelo sin las variables señaladas como no significativas por el tercer modelo. En este último modelo todos los predictores se mostraron estadísticamente significativos y ninguno registró un factor de inflación de la varianza mayor a 1.1.

Para los cuatro modelos ajustados se empleó la función `predict.glm()` para generar un predictor que se usaría para construir la matriz de confusión; tal función tomó como argumento al conjunto de prueba original para los primeros dos modelos, y al conjunto de prueba sin valores atípicos para los modelos restantes. Con ayuda de la función `confusionMatrix()` se obtuvieron la precisión (con su intervalo al 95 % de confianza), sensibilidad y especificidad para cada uno de los clasificadores. Por último se construyó la curva ROC para cada uno de los clasificadores y se obtuvo el AUC. Las métricas mencionadas se observan en la Tabla 8.1 y los residuales de los modelos se observan en la Figura 8.2.

Modelo	Precisión	IC Precisión	Sensib.	Especif.	AIC	Devianza	AUC
Modelo 1	0.7169	(0.677,0.7544)	0.8287	0.6212	1093.938	1069.938	0.8028908
Modelo 2	0.7169	(0.677,0.7544)	0.8287	0.6212	1088.042	1074.042	0.7997634
Modelo 3	0.7524	(0.7129,0.7891)	0.816	0.6917	999.0516	975.0516	0.7937496
Modelo 4	0.7485	(0.7088,0.7854)	0.8088	0.6917	1014.438	1004.438	0.8001874

Tabla 8.1: Comparación del rendimiento de los cuatro modelos de regresión logística ajustados.

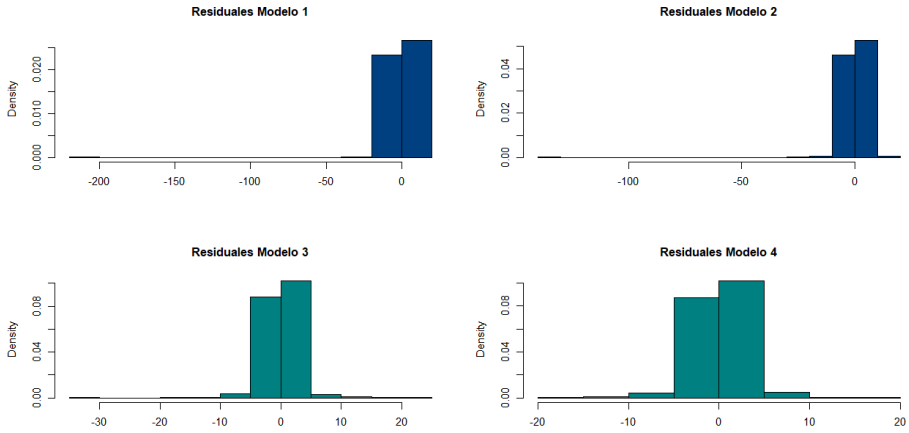

Fig. 8.2: Residuales en la iteración final de cada uno de los ajustes.

8.2. Ajuste del clasificador de Bayes

Para construir el clasificador de Bayes se utilizó la función `naiveBayes()`, donde se especificó como argumento a la variable `Good.quality` explicada por las demás y el conjunto de datos utilizado fue el de entrenamiento original. Este procedimiento se repitió para un segundo clasificador, que utilizó el conjunto de entrenamiento sin datos atípicos. Para ambos clasificadores se obtuvo su matriz de confusión y curva ROC (Figura 8.3), evaluadas en los respectivos conjuntos de prueba de cada uno. Los rendimientos de los clasificadores se pueden ver en la siguiente tabla:

Clasificador	Precisión	IC Precisión	Sensibilidad	Especificidad	AUC
Clasificador I	0.7261	(0.6865,0.7632)	0.7450	0.7099	0.7274588
Clasificador II	0.7447	(0.6949,0.7736)	0.7849	0.7068	0.7276738

Tabla 8.2: Comparación del rendimiento de ambos clasificadores bayesianos.

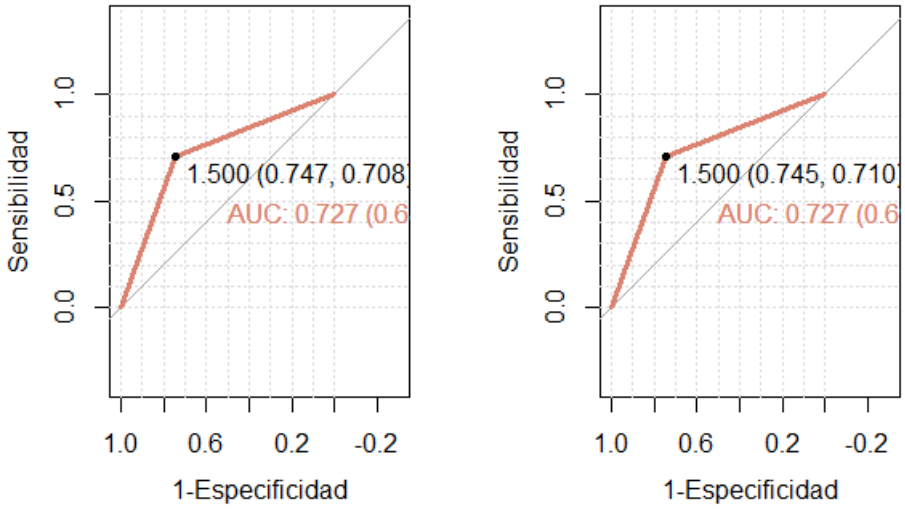

Fig. 8.3: Curva ROC del clasificador de Bayes I (izquierda) y del clasificador de Bayes II (derecha).

8.3. Ajuste con Máquinas de Soporte Vectorial

Como se mencionó en la sección 3.3, los parámetros de las máquinas de soporte vectorial no pueden hallarse de manera directa, sino que deben utilizarse métodos basados en validación cruzada para encontrar los valores óptimos dependiendo del conjunto de datos y el tipo de kernel que se utilice. Para afinar los parámetros se empleó la función `tune()`, dando como argumento una lista arbitraria de valores de los parámetros para evaluar cada ajuste.

Al usar como base el conjunto de datos de entrenamiento original, los parámetros establecidos para los distintos kernels fueron:

- Kernel lineal: $Costo = 0.10$
- Kernel gaussiano: $Costo = 5, \gamma = 0.5$
- Kernel polinomial: $Costo = 0.5, p = 2, \tau = 1$
- Kernel sigmoidal: $Costo = 0.1, \gamma = 0.1, \tau = 0.1$

Los rendimientos se encuentran registrados en la Tabla 8.4, donde se aprecia que la mayor precisión y sensibilidad se obtuvo con el Kernel polinomial.

Kernel	Precisión	IC Precisión	Sensibilidad	Especificidad	AUC
Lineal	0.7261	(0.6865,0.7632)	0.7530	0.7031	0.7280299
Gaussiano	0.7243	(0.6846,0.7614)	0.7052	0.7406	0.7228968
Polinomial	0.7408	(0.7018,0.7772)	0.7530	0.7304	0.7126715
Sigmoidal	0.7096	(0.6694,0.7474)	0.7530	0.6724	0.7126715

Tabla 8.3: Comparación del rendimiento de las máquinas de soporte vectorial con los cuatro tipos de kernel.

El mismo procedimiento se llevó a cabo nuevamente para utilizar los conjuntos de datos sin valores atípicos. Los parámetros ajustados para los kernels fueron:

- Kernel lineal: $Costo = 1.0$
- Kernel gaussiano: $Costo = 5, \gamma = 0.1$
- Kernel polinomial: $Costo = 1, p = 2, \tau = 0.5$
- Kernel sigmoidal: $Costo = 0.1, \gamma = 0.1, \tau = 0.1$

Los rendimientos, al igual que con los modelos anteriores, se registraron en la siguiente tabla:

Kernel	Precisión	IC Precisión	Sensibilidad	Especificidad	AUC
Lineal	0.7118	(0.6707,0.7505)	0.6867	0.7324	0.7095448
Gaussiano	0.7737	(0.7351,0.8091)	0.7768	0.7711	0.7739754
Polinomial	0.7447	(0.7048,0.7817)	0.7124	0.7711	0.7417866
Sigmoidal	0.7060	(0.6647,0.745)	0.7039	0.7077	0.7058046

Tabla 8.4: Comparación del rendimiendo de las máquinas de soporte a partir del segundo conjunto de prueba y entrenamiento.

8.4. Ajuste de árboles CART y bosques aleatorios

Construir un árbol CART es sumamente sencillo en R utilizando la biblioteca `rpart`. Al utilizar la función `rpart()` se puede construir un árbol que toma como argumentos a la variable dependiente explicada por los predictores, el método `class` (utilizado para clasificación) y un

conjunto de datos como base. Esta función construye un árbol y afina sus parámetros únicamente tomando la información disponible del conjunto de datos. Esto resulta útil y rápido en la ejecución, sin embargo, un árbol construido de tal forma puede no ser el mejor para clasificar, o incluso puede caer en circunstancias de sobreajuste. Es por esto que se utilizó la función `train()` con un entrenamiento de validación cruzada para encontrar, a partir del conjunto de entrenamiento, el mejor parámetro de complejidad para la construcción del árbol. La función indicó que el mejor árbol construido (Figura 8.4) sería aquel con un parámetro de complejidad[1] CP = 0.01014199. Al ver la importancia de las variables se encontró que las cuatro más importantes fueron: Alcohol, Sulfitos, Acidez volátil y Cloruros, en ese orden.

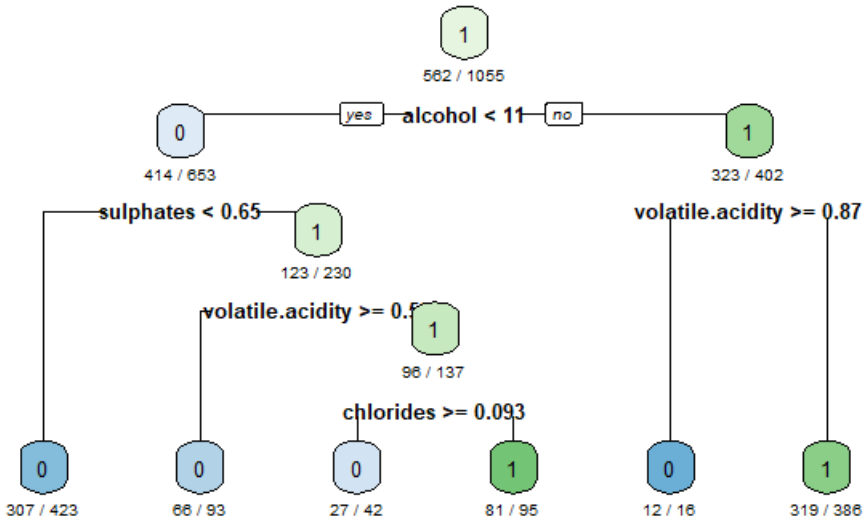

Fig. 8.4: Árbol construido con un CP = 0.01014199. En cada nodo se observa el cociente de las tuplas que caen que pertenecen a la categoría indicada, entre el número total de tuplas registradas. Entre más oscuro sea el color del nodo, mayor es su pureza.

Respecto a los conjuntos de datos sin valores atípicos, se construyó otro árbol (Figura 8.5) con un CP de 0.01287554. Al igual que el árbol anterior, éste mostró exactamente las mismas variables importantes en el mismo orden; no obstante, la construcción del árbol tomó a la variable Dióxido de azufre total. El rendimiento de los árboles se puede ver en la Tabla 8.5.

Árbol	Precisión	IC Precisión	Sensibilidad	Especificidad	AUC
Árbol I	0.7040	(0.6637,0.7421)	0.7729	0.6451	0.7089798
Árbol II	0.6886	(0.6467,0.7283)	0.7382	0.6479	0.6930424

Tabla 8.5: Comparación del rendimiento de los árboles de clasificación.

[1]El parámetro de complejidad se utiliza para la poda del árbol. Un CP con valor uno corresponde a un árbol sin divisiones y un valor igual a cero corresponde a un árbol de profundidad máxima.

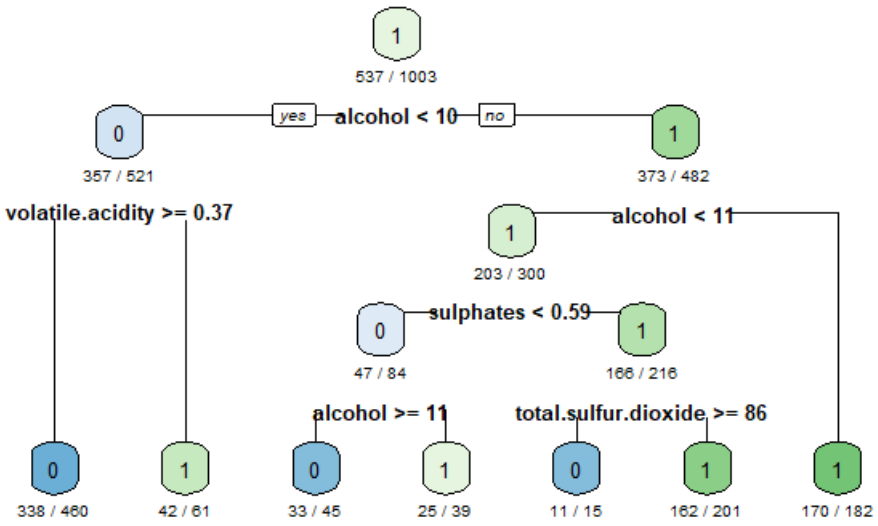

Fig. 8.5: Árbol construido a partir del conjunto de entrenamiento con valores atípicos removidos. A diferencia del anterior, este utiliza al Dióxido de azufre total en lugar de los Cloruros.

Para encontrar los parámetros óptimos del bosque se recurrió nuevamente a la validación cruzada con el conjunto de entrenamiento. La función `train()` mostró que el bosque óptimo tendría 101 árboles y un parámetro mtry[2] igual a seis. Este bosque mostró una precisión de 0.7904, sensibilidad de 0.8048 y especificidad igual a 0.7782 al momento de evaluar con el conjunto de prueba. Las variables más importantes en este modelo fueron: Alcohol, Sulfitos, Acidez volátil y Dióxido de azufre total, en ese orden. Finalmente, un bosque entrenado a partir del conjunto sin valores atípicos, con 63 árboles y mtry igual a tres, mostró un rendimiento todavía mayor al evaluarse en el conjunto de prueba: una precisión de 0.8104, una sensibilidad igual a 0.7940 y una especificidad igual a 0.8239. Este bosque mostró exactamente las mismas variables importantes que el bosque anterior, preservando el mismo orden.

[2]El parámetro mtry se define como el número de variables muestreadas aleatoriamente como candidatas en cada partición.

Capítulo 9

Conclusiones

Como pudimos apreciar al momento de los resultados de los ajustes, todos los clasificadores −salvo los bosques aleatorios− mostraron un rendimiento similar en todas las métricas evaluadas. Esto, por un lado, se debe a la naturaleza del conjunto de datos, ya que pudimos observar que el conjunto por sí mismo era bastante homogéneo y no presentaba problemas extremos con los valores registrados o una dominancia muy marcada de alguna clase. De manera computacional, el tiempo de entrenamiento de los algoritmos también fue similar, salvo los bosques aleatorios y las SVM con kernel polinomial. También se pudo observar que, al menos en este caso en concreto, la existencia de valores atípicos no influyó de manera tan significativa en los rendimientos de los modelos, aunque sí hubieron ligeros cambios en las métricas, no son diferencias tan marcadas. Resultaría atrayente evaluar los modelos con conjuntos de datos heterogéneos, con predictores numéricos y categóricos, con una dimensión mucho más grande en cuanto a número de observaciones o número de predictores.

Los modelos de regresión logística aumentaron su rendimiento sin valores atípicos y además hubo un cambio sustancial en sus residuales (Figura 8.2), lo cual es importante al menos de manera teórica. El modelo 3 presentó el mejor rendimiento a comparación de los demás; no obstante, cabe mencionar que tal modelo mostró problemas de multicolinealidad en los predictores, razón por la que podría presentarse un problema de sobre-ajuste. Recordemos que un sobre-ajuste suele presentarse cuando se estiman más parámetros de los necesarios para un modelo, causando que éste tenga un buen rendimiento con los datos de entrenamiento pero uno malo con los datos de prueba.

En el caso de preferir un modelo sin el problema mencionado, se optaría por el modelo 4. Por otro lado, aunque la implementación de la regresión logística es sencilla, al momento del análisis del modelo ajustado se puede tomar mucho tiempo para la revisión de las variables significativas; esto en conjuntos de datos con muchas variables puede representar un gran contratiempo.

Los clasificadores de Bayes mejoraron un poco sin la presencia de los valores atípicos extremos, aunque no fue un cambio drástico; esto indicaría que este tipo de modelos no se ven influidos por valores extremos. La construcción teórica de estos modelos fue la más sencilla de entender y su implementación también lo fue.

Los modelos basados en máquinas de soporte vectorial fueron bastante complejos de entender puesto que requieren demasiada teoría matemática. El mejor modelo fue el basado en un kernel polinomial, sin embargo, este kernel es el que presenta más parámetros y se tomó mucho tiempo para afinarlos, a comparación de los demás modelos con distintos kernels. Otro problema presentado para estos modelos es que la afinación de los parámetros se hace a partir de una lista de valores arbitraria, lo cual no asegura que alguno de los parámetros existentes en la lista sea el óptimo.

Los árboles fueron sumamente fáciles de entender debido a que su construcción fue intuitiva. Al momento de su implementación y entrenamiento fueron rápidos. Estos modelos presentaron los rendimientos más bajos. Tal situación podría explicarse por mera aleatoriedad; cada vez que se construye un árbol la partición y elección de los atributos cambia, y por tanto cambia su rendimiento. Finalmente, el algoritmo con mejor rendimiento en cualquiera de las métricas fue bosques aleatorios, lo que es congruente debido a su naturaleza más compleja basada en varios árboles.

Bosques aleatorios y los modelos de regresión logística mostraron las mismas variables en cuanto importancia: Alcohol, Sulfitos, Ácidez volátil y Dióxido de azufre total. La elección de estas variables podría resultar interesante si se ajustaran nuevamente los clasificadores bayesianos y las máquinas de soporte vectorial tomando únicamente dichos predictores. Por otro lado, también resultaría interesante ajustar los modelos aquí presentados pero haciendo una pre-selección de variables a partir de una regresión LASSO, o bien, utilizando el método de componentes principales.

Bibliografía

[1] L. J. Sandoval. ALGORITMOS DE APRENDIZAJE AUTOMÁTICO PARA ANÁLISIS Y PREDICCIÓN DE DATOS. *Revista tecnológica (Escuela Especializada en Ingeniería ITCA-FEPADE)*, (11), 2018 (citado en la pág. 1).

[2] L. Rincón. *Introducción a la probabilidad*. L. prensas de ciencias, edición. Ciudad de México, 2.ª edición, 2016 (citado en las págs. 5, 7).

[3] P. Wilmott. *MACHINE LEARNING: An Applied Mathematics Introduction*. P. O. Publishing, edición. 2019 (citado en la pág. 8).

[4] A. C. Müller y S. Guido. *Introduction to Machine Learning with Python*. I. O'Reilly Media, edición. 2017 (citado en las págs. 10, 31, 36).

[5] L. Naranjo Albarrán. Análisis de Regresión Lineal Simple, 2020 (citado en la pág. 12).

[6] M. Durbán. Modelos Lineales Generalizados. URL: http://halweb.uc3m.es/esp/Personal/personas/durban/esp/web/GLM/curso_GLM.pdf (citado en las págs. 14, 17).

[7] A. Martínez Gutiérrez. *Análisis de las Primas de Riesgo en Seguros de Automóviles: Una Aplicación de los Modelos Lineales Generalizados*. Tesis de maestría, Universidad Autónoma Metropolitana, Unidad Iztapalapa, Ciudad de México, 2017 (citado en la pág. 14).

[8] L. Naranjo Albarrán. Modelos lineales generalizados (citado en las págs. 15, 17).

[9] C. Gil Martínez. Regresión logística (simple y múltiple), 2018. URL: https://rpubs.com/Cristina_Gil/Regresion_Logistica (citado en la pág. 18).

[10] E. Campo León. *Introducción a las máquinas de vector de soporte (SVM) en aprendizaje supervisado*, Facultad de Ciencias, Universidad de Zaragoza, Zaragoza, 2016 (citado en la pág. 21).

[11] J. A. Resendiz Trejo. Las maquinas de vectores de soporte para identificación en línea, Ciudad de México, 2006 (citado en las págs. 21, 22).

[12] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer New York, NY, 2.ª edición, 2000 (citado en la pág. 21).

[13] E. D. Monroy Jordan y J. E. Pérez Neira. Máquinas de soporte vectorial (SVM), Cartagena de Indias, 2005 (citado en la pág. 21).

[14] G. A. Betancourt. LAS MÁQUINAS DE SOPORTE VECTORIAL (SVMs). *Scientia et Technica*, (27), 2005. URL: https://www.redalyc.org/articulo.oa?id=84911698014 (citado en la pág. 22).

[15] V. Vapnik y A. Lerner. Pattern recognition using generalized portrait method. Automation and Remote Control. 1963 (citado en la pág. 23).

[16] J. R. Berrendero. Tema 5. Dualidad y condiciones de Karush-Kuhn-Tucker, Madrid (citado en la pág. 25).

[17] C. Cortes y V. Vapnik. Support-vector networks. *Machine Learning*, (20):273-297, 1995 (citado en la pág. 26).

[18] V. Vapnik N. y A. Y. Chervonenkis. The necessary and sufficient conditions for consistency in the empirical risk minimization method. *Pattern Recognition and Image Analysis*, (1):283-305, 1991 (citado en la pág. 27).

[19] M. López García. Notas del minicurso Espacios con núcleo reproductor, 2012 (citado en la pág. 29).

[20] B. E. Boser, G. I. M. y V. Vapnik. A training algorithm for optimal margin classifiers:144-152, 1992 (citado en la pág. 29).

[21] J. L. Ruiz Reina. Evaluación de modelos, 2017,2018 (citado en la pág. 30).

[22] J. Amat Rodrigo. Árboles de decisión, random forest, gradient boosting y C5.0, 2020. URL: https://www.cienciadedatos.net/documentos/33_arboles_de_prediccion_bagging_random_forest_boosting (citado en la pág. 32).

[23] G. Avilés Rosas. Clasificación, 2020 (citado en la pág. 33).

[24] G. Avilés Rosas. Clasificación: Árboles CART, 2021 (citado en la pág. 34).

[25] A. Agresti. *Foundations of Linear and Generalized Linear Models*. John Wiley & Sons, Hoboken, New Jersey, 2015 (citado en la pág. 40).

[26] D. Z. Cárdenas Moreno. *Regresión logística y soluciones para multicolinealidad en riesgo de crédito*, Facultad de Ciencias, UNAM, Ciudad de México, 2021 (citado en la pág. 40).

[27] A. R. del Valle Benavides. *Curvas ROC (Receiver-Operating-Characteristic) y sus aplicaciones*. Tesis doctoral, Universidad de Sevilla, Sevilla, 2017 (citado en la pág. 43).

[28] W. K. Härdle y L. Simar. *Applied Multivariate Statistical Analysis*. Springer, edición. third edición, 2012 (citado en las págs. 46, 47).

[29] R. Salmerón Gómez. Multicolinealidad, 2016 (citado en la pág. 47).

[30] L. Naranjo Albarrán. Regresión Lineal Múltiple, 2020 (citado en la pág. 48).

[31] I. M.-A. Cediel, J. M. de Prádena Lobón, M. G. Mata, M. L. P. Rodríguez, A. R. Cuenca, M. J. V. Suárez y M. A. Z. Revilla. El vino y su análisis, 2014 (citado en las págs. 51, 52).

[32] A. Munoz-Villa, A. S. Galindo, L. L. López, L. Cantú-Sifuentes y L. Barajas-Bermúdez. Ácido Cítrico: Compuesto Interesante. *Revista Científica de la Universidad Autónoma de Coahuila*, **6**(12), 2014 (citado en la pág. 51).

[33] S. de Publicacione Agrícolas. *Análisis de los vinos que han de seguirse por todos los laboratorios dependientes del Ministerio de Agricultura*. Madrid, 3.ª edición, 1934 (citado en la pág. 51).

[34] B. Blondin. La producción de SO_2 por levaduras enológicas durante la fermentación alcohólica, 2015 (citado en la pág. 52).

[35] J. C. C. Chaves, L. G. Rincón y C. A. C. Camargo. REGIONALIZACIÓN HIDROMETEREOLÓGICA: MÉTODO DE CLUSTER Y CURVAS DE ANDREWS, 2019 (citado en la pág. 54).

[36] J. A. Guzmán Luna, J. Cartagena Orrego y J. D. Restrepo Duque. Desarrollo de un sensor móvil para la medición de ruido ambiental, Medellín, 2016 (citado en la pág. 60).