Preparación de los datos

PID_00296763

Jordi Casas Roma
Julià Minguillón Alfonso

Tiempo mínimo de dedicación recomendado: 3 horas

Universitat Oberta de Catalunya

Jordi Casas Roma

Doctor en Informática (Universidad Autónoma de Barcelona, UAB, 2014), máster en Inteligencia Artificial Avanzada (Universidad Nacional de Educación a Distancia, UNED, 2011) y licenciado en Informática (Universidad Autónoma de Barcelona, UAB, 2002). Hasta 2023 ha sido profesor agregado de la Facultad de Informática, Multimedia y Telecomunicaciones de la Universitat Oberta de Catalunya (UOC). Desde 2024 es profesor lector del Departamento de Informática de la Universitat Autònoma de Barcelona (UAB). Sus actividades docentes se concentran principalmente en minería de datos, aprendizaje automático, *big data* y minería de gráficos. Sus principales ámbitos de investigación incluyen la inteligencia artificial, el aprendizaje automático, la minería de grafos y la privacidad de datos. Desde el 2020 es líder de grupo a ADaS Lab (Applied Data Science Lab) donde investiga la aplicación de algoritmos de inteligencia artificial a problemas del mundo real.

Julià Minguillón Alfonso

Profesor de la Universitat Oberta de Catalunya (UOC) desde 2001. Centra su actividad de investigación en el análisis y desarrollo de herramientas y recursos educativos abiertos en entornos virtuales de aprendizaje. Doctor por la Universitat Autònoma de Barcelona (UAB) con una tesis doctoral en el ámbito del aprendizaje automático. Sus líneas de trabajo se centran en el desarrollo de aplicaciones para la gestión, el análisis y la visualización de datos provenientes de entornos de aprendizaje virtuales, como el Campus Virtual de la UOC, y de redes sociales, especialmente Wikipedia. Es autor de diversos recursos docentes del ámbito de la ciencia de datos relacionados con la programación y el análisis y visualización de datos. Desde 2024 es el director del programa del máster universitario de Ciencia de Datos de la UOC.

Cómo citar este recurso de aprendizaje con el estilo Harvard:
Casas Roma, J.; Minguillón Alfonso, J. (2024). *Preparación de los datos*. [Recurso de aprendizaje textual]. 1.ª ed. Barcelona: Fundació Universitat Oberta de Catalunya (FUOC).

Primera edición: septiembre 2024

Av. Tibidabo, 39-43, 08035 Barcelona
Autoría: Jordi Casas Roma, Julià Minguillón Alfonso
Producción: FUOC

Índice general

Introducción

En este módulo didáctico se presentan los conceptos y técnicas básicas para el preprocesamiento de datos. Se trata de una etapa crucial en el ciclo de desarrollo de modelos de aprendizaje automático. Su objetivo principal es preparar y limpiar los datos antes del proceso de entrenamiento de un modelo de aprendizaje automático. Esta preparación de datos es esencial para mejorar la calidad de los datos, eliminar ruido, corregir inconsistencias y garantizar que los datos sean adecuados y útiles para el entrenamiento y la evaluación de los modelos.

La importancia del preprocesamiento de datos radica en que los modelos de aprendizaje automático son altamente sensibles a la calidad de los datos de entrada. En consecuencia, un conjunto de datos mal preparado o con características no relevantes puede llevar a un rendimiento deficiente del modelo, un sobreajuste o un sesgo indeseado. Esto es lo que se conoce como GIGO, acrónimo de *Garbage In, Garbage Out*.

Objetivos

En los materiales didácticos de este módulo encontraremos las herramientas indispensables para asimilar los siguientes objetivos:

1. Comprender los procesos y técnicas relevantes para la preparación de los datos, incluyendo técnicas de limpieza, discretización, normalización y estandarización de datos.
2. Conocer los principales métodos de reducción de la dimensionalidad.
3. Conocer las principales técnicas de extracción y selección de atributos o características.
4. Conocer las principales técnicas que permiten reducir la problemática de trabajar con conjuntos de datos desbalanceados.

1. Contextualización

El preprocesamiento de datos es una etapa crucial en el ciclo de desarrollo de modelos de aprendizaje automático. Su objetivo principal es preparar y limpiar los datos antes del proceso de entrenamiento de un modelo de aprendizaje automático. Esta preparación de datos es esencial para mejorar la calidad de los datos, eliminar ruido, corregir inconsistencias y garantizar que los datos sean adecuados y útiles para el entrenamiento y la evaluación de los modelos.

La importancia del preprocesamiento de datos radica en que los modelos de aprendizaje automático son altamente sensibles a la calidad de los datos de entrada. En consecuencia, un conjunto de datos mal preparado o con características no relevantes puede llevar a un rendimiento deficiente del modelo, un sobreajuste o un sesgo indeseado.

En relación con la notación de este texto, emplearemos la siguiente notación para referirnos al conjunto de entrada de datos:

$$D_{n,m} = \begin{pmatrix} d_1 = & a_{1,1} & a_{1,2} & \cdots & a_{1,m} & c_1 \\ d_2 = & a_{2,1} & a_{2,2} & \cdots & a_{2,m} & c_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ d_n = & a_{n,1} & a_{n,2} & \cdots & a_{n,m} & c_n \end{pmatrix}$$

donde:

- n es número de elementos o instancias,
- m el número de atributos del conjunto de datos o también su dimensionalidad,
- d_i indica cada una de las instancias del juego de datos,
- $a_{i,j}$ indica cada uno de los atributos descriptivos, y
- c_i indica el atributo objetivo o clase a predecir para cada elemento, suponiendo que solamente hay una clase para cada elemento.

Es importante destacar que el atributo objetivo o clase a predecir (c_i) únicamente existe (o aplica) en los métodos supervisados. En el caso de los méto-

dos no supervisados, el conjunto de datos sigue el mismo patrón y notación, excepto para el atributo de clase (c_i) que no existe (o no se emplea) en el caso de los conjuntos no etiquetados.

2. Conceptos preliminares

En este apartado revisaremos algunos conceptos matemáticos que son especialmente útiles en los algoritmos de minería de datos, y que, por lo tanto, es interesante conocer antes de iniciar el estudio de los diferentes métodos o algoritmos.

En primer lugar, veremos muy brevemente los conceptos más básicos del análisis estadístico, en el subapartado 2.1. A continuación, en el subapartado 2.2 repasaremos algunas de las métricas de distancia o similitud más empleadas por los algoritmos de minería de datos. Finalizaremos este repaso de conceptos preliminares con una breve reseña de la entropía y la ganancia de información, en el subapartado 2.3.

2.1. Análisis estadístico

La estadística mira los datos y trata de analizarlos, por este motivo la minería de datos utiliza tantos conceptos que provienen de este campo de conocimiento. La recogida de muestras, la exploración de los datos, la inferencia estadística y la búsqueda de patrones forman parte de ambas disciplinas.

2.1.1. La distribución de una muestra

A continuación veremos las bases estadísticas que pueden ayudarnos en la minería de datos, centrándonos en el concepto de distribución con el objetivo de introducirnos en el ámbito de conocimiento de la estadística muestral.

Principales estimadores

Alfabeto griego y latino

Normalmente se utiliza el alfabeto griego para hacer referencia a la población y el latino para la muestra.

La estadística entiende los datos como **observaciones** de una **variable**. Generalmente, se suele trabajar con el alfabeto griego y latino más que con números, de modo que se suele representar una variable con la mayúscula X y sus observaciones con la minúscula $X = \{x_1, x_2, \ldots, x_n\}$.

La **distribución de una variable** son las propiedades de los valores observados, como por ejemplo los valores mayor y menor o los valores con mayor o menor frecuencia de aparición. Un aspecto interesante en una distribución

es el estudio de su zona central. Es decir, los valores que aproximadamente tienen la mitad de las observaciones por debajo (con menos frecuencia) y la mitad por arriba (con más frecuencia). Aquí es donde encontramos los dos estimadores más básicos: la mediana y la media aritmética.

La **mediana** (*median*) corresponde al valor central de la distribución, que podemos expresar como

$$M_e = x_{\frac{n+1}{2}} \tag{1}$$

donde n es el número total de observaciones. Si n es par, entonces la mediana es

$$M_e = \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n+1}{2}}) \tag{2}$$

La **media aritmética** (*arithmetic mean*) de un conjunto de datos numéricos es su valor medio, representado por:

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{3}$$

Tanto la mediana como la media aritmética miden el centro de una distribución, pero lo hacen de forma distinta. Solo cuando la distribución es simétrica, ambos valores coinciden. Los valores asimétricos y alejados atraen (es decir, sesgan) la media, mientras que no afectan tanto a la mediana. Esto hace que la media por sí sola no sea un estimador suficiente para describir la distribución de una variable.

La media aritmética, en ocasiones, puede llegar a ser muy poco representativa del juego de datos. Por este motivo suele ir acompañada de la varianza, donde una varianza pequeña indica que el juego de datos es compacto y en consecuencia la media será muy representativa.

La **varianza**, entendida como la suma de los cuadrados de las distancias a la media, permite medir la dispersión alrededor de la media aritmética. Se calcula de la siguiente forma:

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1} \tag{4}$$

Es importante observar que, a diferencia del cálculo de la media, en este caso se divide por $n-1$ y no por n. Es lo que se conoce como la corrección de Bessel, y permite corregir el sesgo estadístico en la estimación de la varianza poblacional, y algunos (pero no todos) los sesgos en la estimación de la desviación estándar poblacional.

La **desviación estándar** es simplemente la raíz cuadrada de la varianza, de modo que nos da una medida de la dispersión en torno a la media, pero expresada en la misma escala que la variable. Su fórmula de cálculo es obviamente:

$$s = \sqrt{s^2} \tag{5}$$

La importancia de la desviación estándar, como medida de dispersión de la distribución de una variable, radica en el hecho de que "la mayoría" de las desviaciones respecto de la media caen dentro de un intervalo igual a 1 o 2 veces la desviación estándar, es decir, $(\overline{x}-s,\overline{x}+s)$ o $(\overline{x}-2s,\overline{x}+2s)$, como se puede ver en la figura 1.

Mayoría

El concepto de "la mayoría" lo definiremos con el estudio de la distribución normal.

La distribución normal

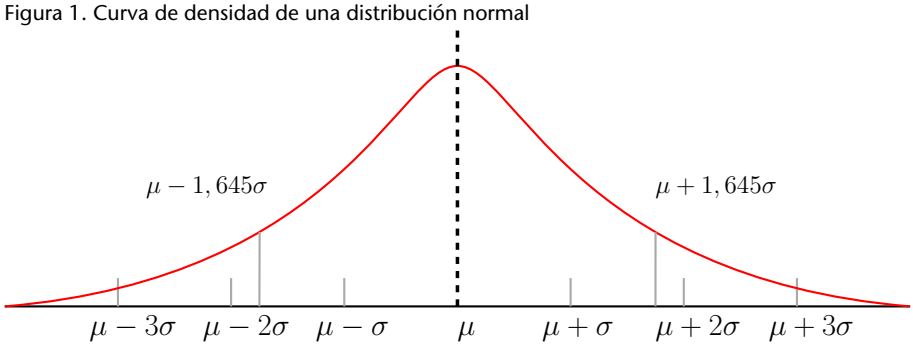
La **curva de densidad** es la representación gráfica de la distribución de una variable, suponiendo que fuéramos capaces de obtener muchísimas observaciones. Una **distribución normal** cumple la propiedad de que su curva de densidad es aproximadamente simétrica, como se puede ver en la figura 1, y por lo tanto la mediana y la media aritmética coinciden en su centro. La fórmula matemática de la distribución normal viene dada por:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{6}$$

Terminología

La distribución normal también es conocida como distribución de Gauss o distribución gaussiana.

Figura 1. Curva de densidad de una distribución normal



Debido a las propiedades especiales que posee la distribución normal, reservamos letras específicas para referirnos a su media y a su desviación estándar:

- La letra μ se utiliza para referirse a la media aritmética de una distribución normal.
- La letra σ se utiliza para referirse a la desviación estándar de una distribución normal.

Si nos fijamos en estos intervalos, sucede que:

- En toda distribución normal, el área debajo de la curva para el intervalo de una desviación estándar es de aproximadamente 0.68. Es decir, en el intervalo $(\mu - \sigma, \mu + \sigma)$ encontraremos el 68 % de la población.
- En el intervalo $(\mu - 1{,}645\sigma, \mu + 1{,}645\sigma)$ encontramos el 90 % de la población.
- En el intervalo $(\mu - 2\sigma, \mu + 2\sigma)$ encontraremos el 95 % de la población.
- Y finalmente, en el intervalo $(\mu - 3\sigma, \mu + 3\sigma)$ encontraremos el 99,7 % de la población.

La distribución normal estándar

La **distribución normal estándar** es una distribución normal que tiene una media aritmética con valor 0 y una desviación estándar con valor igual a 1.

Es interesante, dada una variable, poder **estandarizarla**, es decir, realizar los cálculos de probabilidades (áreas debajo de la curva de densidad) necesarios y regresar de nuevo a la variable original con la traducción de las probabilidades calculadas. Para realizar este proceso, partiremos de una variable X normalmente distribuida, con media aritmética μ y desviación estándar σ, y la estandarizaremos restando a cada observación la media aritmética y dividiendo el resultado por la desviación estándar. El resultado será la variable **variable normal estandarizada**, que se calcula de la siguiente forma:

$$Z = \frac{X - \mu}{\sigma} \tag{7}$$

Si llamamos P a la probabilidad o área debajo de la curva de densidad, consultando las tablas de densidad de la distribución normal estándar sabemos que, al ser una distribución simétrica centrada en el cero, se cumple que:

$$P(Z < -1) = P(Z > 1) = 0{,}1587 \tag{8}$$

y por deducción, como toda el área bajo la curva de dicha distribución es 1, podemos calcular la probabilidad de que un valor de Z se encuentre en la zona central, que corresponde a:

$$\begin{aligned} P(-1 < Z < 1) &= 1 - P(Z < -1) - P(Z > 1) \\ &= 1 - 2 * 0{,}1587 = 0{,}6826 \end{aligned} \tag{9}$$

De ahí que aproximadamente el 68 % de la población se encuentre entre -1 y 1, o entre –σ y σ en general. Para realizar el camino inverso, es decir, de la distribución normal estándar a la distribución normal original, usaremos la siguiente expresión:

$$X = \mu + Z\sigma \tag{10}$$

2.2. Métricas de distancias o similitud

Antes de adentrarnos en el subapartado de algoritmos, nos centraremos en los conceptos de distancia y similitud (o disimilitud), puesto que estos constituyen la base sobre la que se construyen muchos de los algoritmos de segmentación y clasificación.

La elección de una métrica apropiada para la distancia tiene mucho impacto en el resultado final de cualquier algoritmo de segmentación. Es más, la elección de la métrica de distancia adecuada suele ser uno de los aspectos clave en la construcción de este tipo de algoritmos.

Cuando hablamos de distancia, en realidad nos estamos refiriendo a una forma de cuantificar cuán similares (o disimilares) son dos objetos, variables o puntos.

Planteado de esta forma, el concepto de distancia es muy abstracto y etéreo, por este motivo los científicos quieren ponerle algunos límites o condiciones. Para un conjunto de elementos X se considera distancia a cualquier función $f : X \times X \to \mathbb{R}$ que cumpla las siguientes tres condiciones:

- **No negatividad**: la distancia entre dos puntos siempre debe ser positiva.

$$d(a,b) \geq 0 \quad \forall a,b \in X \tag{11}$$

- **Simetría**: la distancia entre un punto a y un punto b debe ser igual a la distancia entre el punto b y el punto a.

$$d(a,b) = d(b,a) \quad \forall a,b \in X \tag{12}$$

- **Desigualdad triangular**: la distancia debe coincidir con la idea intuitiva que tenemos de ella en cuanto a que es el camino más corto entre dos puntos.

$$d(a,c) \leq d(a,b) + d(b,c) \quad \forall a,b,c \in X \tag{13}$$

Veamos, a continuación, algunas de las métricas de distancia o similitud más empleadas por los algoritmos de minería de datos.

Distancia euclídea

La distancia euclídea o euclidiana es una de las distancias más utilizadas, especialmente en el caso de tratar con atributos numéricos. La distancia euclídea coincide plenamente con lo que nuestra intuición entiende por "distancia". Su expresión matemática para un espacio de m dimensiones es:

$$d(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2} \tag{14}$$

Uno de sus principales inconvenientes se presenta al utilizar esta distancia en un proceso de segmentación, donde este cálculo de la distancia no tiene en cuenta las diferentes unidades de medida en las que pueden estar expresadas las variables x e y. Por ejemplo, si estamos segmentando familias o razas de ratones, donde la variable x representa la edad de cada ratón y la variable y representa la longitud de la cola del ratón (en milímetros), entonces la variable x tomará valores del rango [0,3], mientras que la variable y utilizará el rango [90,120]. Parece claro que cuando calculemos la distancia o similitud entre dos individuos, pesará injustamente mucho más la longitud de la cola que la edad.

Distancia estadística o de Gauss

Para superar la distorsión provocada por las diferentes unidades de medida usadas en las distintas variables, tenemos la distancia estadística, que simplemente "normaliza" las variables para situarlas todas bajo la misma escala.

Su expresión analítica para un espacio de m dimensiones, en la que nuestra distribución de puntos en estas dimensiones tuviera una desviación estándar σ viene dada por la fórmula:

$$d(x,y) = \sqrt{\sum_{i=1}^{m}\left(\frac{x_i - y_i}{\sigma_i}\right)^2} \tag{15}$$

Nuevamente, este concepto de distancia sigue teniendo problemas. No tiene en cuenta la correlación entre las variables, es decir, si nuestras variables de trabajo fueran totalmente independientes, no habría ningún problema, pero si tienen algún tipo de correlación, entonces una influye sobre la otra y esta "influencia" no queda bien reflejada si usamos la distancia estadística.

Por ejemplo, las variables *entrenamiento* y *rendimiento* están correlacionadas, de modo que más entrenamiento siempre implica más rendimiento, pero esta regla no se cumple infinitamente, ya que entrenamiento infinito no implica rendimiento infinito (lo vemos continuamente en el deporte). Si no tenemos en cuenta esta correlación y queremos comparar a dos deportistas, podríamos llegar a conclusiones erróneas.

Distancia de Mahalanobis

Esta métrica pretende corregir la distorsión provocada por la correlación de las variables mediante la siguiente expresión para un espacio de m dimensiones:

$$d(x,y) = \sqrt{(x_1 - y_1, \ldots, x_m - y_m) \begin{pmatrix} Cov(x_1,y_1) & \ldots & Cov(x_1,y_m) \\ \vdots & \ddots & \vdots \\ Cov(x_m,y_1) & \ldots & Cov(x_m,y_m) \end{pmatrix}^{-1} \begin{pmatrix} x_1 - y_1 \\ \vdots \\ x_m - y_m \end{pmatrix}} \quad (16)$$

donde $Cov(x,y)$ representa la covarianza entre las variables x e y.

La distancia de Mahalanobis es una generalización de la distancia de Gauss, la cual, a su vez, es una generalización de la distancia euclídea:

- Si en la distribución de puntos las m dimensiones son totalmente independientes, tendremos que su covarianza es 0, de manera que la distancia de Mahalanobis coincide con la distancia estadística.

- Si en la misma distribución con variables independientes tenemos que su distribución está normalizada, es decir, que las dos variables tienen la misma escala, entonces su varianza es 1, de manera que la distancia estadística coincide con la distancia euclídea.

Distancia de Hamming

Hasta ahora hemos revisado algunas de las distancias más importantes para lidiar con valores numéricos. En el caso de tratar con atributos nominales o binarios, una de las métricas más utilizadas es la distancia de Hamming. Esta métrica se define como el número de atributos distintos que existen entre dos vectores de atributos.

$$d(x,y) = \sum_{i=1}^{m} \delta(x_i,y_i) \tag{17}$$

donde $\delta(x_i,y_i)$ toma el valor 0 si $x_i = y_i$ y 1 en caso contrario.

Existen muchas otras funciones de distancia, las cuales intentan capturar la naturaleza de los datos. Os recomendamos darle un vistazo al paquete *philentropy** de R, que permite calcular diferentes funciones de distancia agrupadas por familias.

Lectura complementaria

El lector interesado en otras métricas posibles adaptadas a la naturaleza de los datos puede consultar el trabajo: Cha, S.-H. (2007). "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions". International Journal of Mathematical Models and Methods in Applied Sciences, 1, 300-307.

*** Véase https://cran.r-project.org/web/packages/philentropy/vignettes /Introduction.html**

2.3. Entropía y ganancia de información

En la base conceptual de varios métodos de minería de datos se encuentra el concepto de ganancia de la información, como hilo heurístico que puede guiar el proceso de aprendizaje.

Empezaremos por introducir la idea de **entropía** (*entropy*) como medida de cómo de predecible es un resultado en un juego de datos. También puede ser pensada como el grado de desorden o de incertidumbre presente en un juego de datos. Su expresión matemática es la siguiente:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{18}$$

donde X es una variable aleatoria discreta con un conjunto de posibles valores $X = \{x_1, \ldots, x_n\}$ y $p(x_i)$ es la probabilidad de que la variable X tome el valor x_i.

Valores de entropía altos indican que el resultado es muy aleatorio y, en consecuencia, poco predecible. Veamos, como ejemplo, los siguientes experimentos:

- Tirar un dado con seis caras al aire puede darnos seis posibles resultados, $X \in \{1,2,3,4,5,6\}$. La entropía de este experimento es:

$$H(X) = -6 \cdot \left(\frac{1}{6} \log_2 \left(\frac{1}{6}\right)\right) \approx 2{,}58$$

- Tirar una moneda al aire puede darnos dos posibles resultados, $X \in \{cara,cruz\}$. La entropía de este experimento es:

$$H(X) = -2 \cdot \left(\frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1$$

Observamos como la entropía de lanzar un dado con seis caras es mucho más alta y, por lo tanto, mucho más aleatoria que la entropía de lanzar una moneda. Dicho de otra forma, es más fácil acertar un lanzamiento de cara o cruz que el de un dado con seis caras.

La **ganancia de información** (*information gain*) nos da una medida de cuán relevante es un atributo dentro de un juego de datos, de modo que un atributo con mucha ganancia será muy relevante en el juego de datos, es decir, muy determinante para predecir el atributo objetivo o clase.

La ganancia de información refleja el cambio en la entropía del juego de datos cuando tomamos parte de la información como dada. Dicho de otro modo, la ganancia de información del atributo a respecto del atributo objetivo x es la diferencia entre la entropía del atributo objetivo y la entropía del atributo a:

$$IG(X,a) = H(X) - H(X|a) \tag{19}$$

$$H(X|a) = \sum_{v \in valores(a)} p(v)H(\{x \in X \mid valor(x,a) = v\}) \tag{20}$$

donde $H(X|a)$ es la entropía del juego de datos cuando lo particionamos en base al atributo a.

En la expresión anterior, el valor $p(v)$ es la probabilidad de que el atributo a tenga el valor v y $H(\{x \in X \mid valor(x,a) = v\})$ es la entropía del atributo a cuando toma el valor v en el juego de datos X, es decir, que el valor del atributo a sea v.

El atributo que nos proporcione la mayor ganancia de información (es decir, reduzca la entropía) es candidato a ser usado para clasificar o segmentar el conjunto de datos en diferentes subconjuntos.

3. Preparación de los datos

Las tareas o técnicas de preparación de datos en un proyecto de minería de datos se orientan a la adecuación del juego de datos para que pueda ser usado, posteriormente, por algoritmos de clasificación, segmentación o regresión.

Estas tareas las podemos agrupar en los siguientes bloques temáticos:

- Tareas de **limpieza de datos**, que permiten corregir o eliminar ruido o datos no válidos.
- Tareas de **normalización de datos**, que facilita la presentación de los datos en el mismo rango.
- Tareas de **discretización**, entendidas como procesos de conversión de variables continuas a categóricas.
- Tareas de **reducción de la dimensionalidad**, que nos ayudarán a desarrollar modelos con juegos de datos reducidos.

3.1. Limpieza de datos

En el proceso de limpieza de datos (en inglés, *data cleansing* o *data scrubbing*) se llevan a cabo actividades de detección, eliminación o corrección de instancias corrompidas o inapropiadas en los juegos de datos.

A nivel de valores de atributos se gestionan los valores ausentes, los erróneos y los inconsistentes. Un ejemplo podrían ser los valores fuera de rango, también llamados *(outliers)*.

El proceso de integración de datos puede ser una de las principales fuentes de incoherencias en los datos, ya que fruto de la fusión de juegos de datos distintos se pueden generar inconsistencias que deben ser detectadas y subsanadas.

3.2. Normalización y estandarización de datos

Normalización y estandarización son dos técnicas comunes utilizadas para ajustar las características o variables de un conjunto de datos con el objetivo de mejorar el rendimiento y la precisión de los algoritmos de aprendizaje au-

tomático. Ambas técnicas son aplicadas en el preprocesamiento de datos, que es una etapa crucial para preparar los datos antes de alimentarlos a los modelos de aprendizaje automático, para evitar que ciertas características dominen sobre otras debido a sus escalas originales.

La normalización y la estandarización son técnicas que comparten el objetivo de ajustar las características de un conjunto de datos de tipo numérico, pero difieren en la forma en que lo hacen y en su sensibilidad a los valores atípicos.

En esencia, distinguimos entre dos formas de realizar este proceso:

- **Normalización**: consiste en reescalar las características en un rango específico, generalmente entre 0 y 1. Para normalizar nuestros datos, simplemente necesitaremos aplicar un método de escalado que permita obtener los valores de cada columna de características dentro del rango indicado.

- **Estandarización**: consiste en transformar las características para que tengan una media de 0 y una desviación estándar de 1. Se realiza restando la media de la característica y dividiendo, después, por su desviación estándar.

3.2.1. Normalización

La normalización de datos consiste en modificar los datos para lograr que estén en una escala de valores equivalentes que simplifique la comparación entre ellos, generalmente en el rango [0,1]. La normalización es útil para varios métodos de minería de datos, que tienden a quedar sesgados por la influencia de los atributos con valores más altos, distorsionando de esta forma el resultado del modelo, como por ejemplo pasa con la distancia euclídea.

Algunos de los principales métodos de normalización de atributos son:

- **Normalización por el máximo**, que consiste en encontrar el valor máximo del atributo a normalizar y dividir el resto de los valores por este valor máximo, asegurándonos, por tanto, de que el máximo recibe el valor 1 y el resto valores en el rango [0,1].

$$z_i = \frac{x_i}{x_{max}} \tag{21}$$

  donde z_i es el valor normalizado, x_i el valor original del atributo y x_{max} el valor máximo para este atributo de todo el conjunto de datos.

- **Normalización por la diferencia**, que intenta compensar el efecto de la distancia del valor que tratamos con respecto al máximo de los valores observados. Su fórmula de cálculo es:

$$z_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (22)$$

donde x_{min} y x_{max} son, respectivamente, el valor mínimo y el valor máximo para este atributo en el conjunto de datos.

Uno de los principales puntos débiles de la normalización es su sensibilidad a los valores atípicos (*outliers*), debido a que se basa en el rango máximo y mínimo de la característica. Los valores atípicos pueden afectar significativamente la escala de los datos normalizados.

3.2.2. Estandarización

La estandarización de datos consiste en transformar las características para que tengan una media de cero y una desviación estándar de uno, de forma que las columnas de características tengan los mismos parámetros que una distribución normal estándar (media cero y varianza unidad). De esta forma, es mucho más fácil para los algoritmos "aprender" los pesos de los parámetros independientemente de su rango y escala. Adicionalmente, mantiene información útil sobre valores atípicos y permite a los modelos ser menos sensibles a ellos.

Tanto la normalización como la estandarización son ampliamente utilizadas en problemas de *machine learning*, especialmente la segunda. El método de normalización de atributos más habitual es:

- **Normalización basada en la desviación estándar**, también llamada estandarización de valores, que asegura que se obtienen valores dentro del rango elegido que tienen como propiedad que su media es el cero y su desviación estándar es uno.

$$z_i = \frac{x_i - \mu}{\sigma} \quad (23)$$

donde μ representa la media y σ la desviación típica.

La estandarización es menos sensible a los valores atípicos que la normalización, ya que se centra en la media y la desviación estándar de la característica, lo que reduce el impacto de los valores atípicos en la escala.

3.3. Discretización de datos

La discretización es el proceso mediante el cual los valores de una variable continua se incluyen en contenedores, intervalos o grupos, con el objetivo de

que haya un número limitado de estados o valores posibles. Los contenedores se tratarán entonces como si fueran categorías representadas por un valor.

La discretización es una tarea habitual en los procesos de minería de datos, puesto que muchos algoritmos de clasificación requieren que sus atributos sean discretizados, bien porque solo acepten valores nominales, bien porque trabajar con atributos nominales en lugar de continuos disminuye el coste computacional y acelera el proceso inductivo. Además, la discretización de atributos puede, en ocasiones, mejorar el rendimiento de un clasificador, entre otras razones porque captura mejor la naturaleza del problema a resolver. Por ejemplo, la edad de un usuario puede ser agrupada en franjas de edad 0-12, 13-18, 18-25, etc., de forma que se pueden plantear hipótesis relacionadas con el conocimiento sobre cada grupo de edad.

Los métodos de discretización se pueden clasificar según tres grandes categorías:

- **Supervisados o no supervisados**. Es decir, si los métodos tienen en cuenta los valores del atributo clase o no.

- **Locales o globales**. Métodos limitados a un atributo, o, en general, a un subconjunto horizontal (subconjunto de instancias) o vertical (subconjunto de atributos) del conjunto de datos; o bien actuando sobre todos los atributos y todos los datos a la vez.

- **Parametrizados y no parametrizados**. Los primeros empiezan conociendo el número máximo de intervalos que hay que generar para un atributo específico, mientras que los demás intentan encontrar este número automáticamente.

3.3.1. Método de igual amplitud

El método de partición en intervalos de la misma amplitud es un método no supervisado que busca crear grupos con el mismo número de instancias en su interior.

El método consiste en los siguientes pasos:

1) Se parte de un atributo con n valores, el cual queremos discretizar en k intervalos de igual longitud.

2) La longitud de cada intervalo será:

$$\alpha = \frac{x_{max} - x_{min}}{k} \tag{24}$$

donde x_{min} y x_{max} representan el valor mínimo y máximo, respectivamente, que toman los atributos.

3) Entonces, el inicio del intervalo i de anchura α queda definido por la expresión:

$$Intervalo_i = x_{min} + i\alpha, \forall i \in \{1, \ldots, k\} \quad (25)$$

Algunos de los principales problemas de este método son:

1) Da la misma importancia a todos los valores, independientemente de su frecuencia de aparición.

2) Puede generar intervalos poco homogéneos, donde se mezclen dentro de un mismo intervalo valores que corresponden a clases diferentes.

3) La elección del parámetro k no es trivial en la mayoría de casos.

3.3.2. Método de igual frecuencia

Este método es similar al método de igual amplitud, pero en este caso se requiere que cada grupo contenga el mismo número de instancias en su interior.

El método consiste en los pasos siguientes:

1) Ordenar los valores del atributo X de menor a mayor: $x_1 \leq \ldots \leq x_n$.

2) Fijar un número de intervalos k.

3) Calcular el valor de la frecuencia de cada intervalo: $f = \frac{n}{k}$.

4) Asignar a cada intervalo los f valores ordenados, de tal forma que cada cada intervalo o grupo contendrá f elementos ordenados, donde el valor máximo del intervalo i será menor o igual que el valor mínimo del intervalo $i + 1$.

El principal inconveniente de este método es que valores muy dispares pueden ir a parar al mismo intervalo, dado que el criterio es mantener la misma frecuencia en todos los intervalos. Por el contrario, el método funciona correctamente cuando se da una distribución uniforme de los valores del atributo a discretizar.

3.3.3. Método *chi merge*

Chi merge es un algoritmo simple que utiliza el estadístico chi-cuadrado para discretizar los atributos numéricos. Se trata de un método de discretización de datos supervisado.

Este método intenta que dentro de cada intervalo aparezcan representadas todas las clases de forma parecida, y muy diferente respecto de otros intervalos. En términos de frecuencias, eso quiere decir que dentro de un intervalo hay que esperar que la frecuencia de aparición de valores de cada clase sea parecida, y muy diferente de la de otros intervalos.

Si la primera condición no se cumple, quiere decir que tenemos un intervalo demasiado heterogéneo y que sería preciso subdividirlo. Si no se cumple la segunda, entonces quiere decir que el intervalo es muy parecido a algún otro intervalo adyacente, de manera que, en tal caso, sería necesario unirlos.

Una manera de medir estas propiedades es recurriendo al estadístico chi-cuadrado χ^2 para averiguar si las frecuencias correspondientes a dos intervalos son significativamente diferentes (y, por lo tanto, ya es correcto que los intervalos estén separados), o bien que no lo son (y hay que unirlos).

3.3.4. Métodos basados en entropía

Los métodos basados en medidas de entropía parten de la idea de encontrar particiones del conjunto original de datos que sean internamente tan homogéneas como sea posible, y tan diferentes como sea posible con respecto al resto de las particiones.

Estos métodos trabajan sobre variables continuas con el objetivo de crear un punto de decisión (llamado límite o umbral) sobre un atributo determinado tal que permita alcanzar la mejor separación de los datos en dos subconjuntos disjuntos.

Se pueden definir distintos criterios para escoger la partición "óptima" en cada iteración del algoritmo, pero una de las métricas más empleadas para tal fin es la entropía de la información asociada a cada posible partición. Es decir, se toma como criterio de discretización el valor que minimice la entropía de los subconjuntos generados utilizando ese punto de corte.

3.4. Reducción de la dimensionalidad

En algunos procesos de minería de datos podemos encontrarnos con el problema de que el método de construcción del modelo elegido no puede tratar la cantidad de datos de que disponemos, o simplemente que el tiempo requerido para construir el modelo no es adecuado para nuestras necesidades. Muchos algoritmos tienen una complejidad no lineal (cuadrática, cúbica, exponencial...), lo cual los hace inusables si la dimensionalidad de los datos es demasiado elevada.

En estos casos, es posible aplicar una serie de operaciones con el fin de facilitar dos objetivos, y asegurando, además, que se mantiene la calidad del modelo resultante:

- la **reducción del número de atributos** a considerar,
- y la **reducción del número de casos** que hay que tratar.

Si denotamos como $D_{n,m}$ a nuestro conjunto de datos, en el primer caso el objetivo será reducir el valor de m (reducción del número de columnas), mientras que en el segundo caso el objetivo será reducir el valor de n (reducción del número de filas).

3.4.1. Reducción del número de atributos

La reducción del número de atributos consiste en encontrar un subconjunto de los atributos originales que permita obtener modelos de la misma calidad que los que se obtendrían utilizando todos los atributos, pero con una reducción en la complejidad temporal y/o de cálculo necesaria para el modelo de minería de datos usado posteriormente.

Existen dos grandes grupos:

- Los **métodos de selección de atributos**, que consisten en saber qué subconjunto de atributos puede generar un modelo con la misma (o similar) calidad que el modelo obtenido con el conjunto de atributos original. En esencia, se trata de encontrar métricas que indiquen los atributos que pueden ignorarse en la creación del modelo sin sufrir una considerable pérdida de información.

  Algunos métodos realizan este tipo de pruebas a cada atributo de forma individual, como por ejemplo la prueba de significación, que compara si un atributo es relevante o no a partir de sus valores. Otros métodos, en cambio, parten de que la suposición de independencia entre los atributos puede ser un poco excesiva, y analizan las dependencias y correlaciones entre atributos para determinar grupos de atributos.

- La **fusión y creación de nuevos atributos**, que consiste en crear nuevos atributos "híbridos" a partir de la integración o fusión de los atributos existentes. En general, podemos crear un atributo nuevo a_{n+1} mediante la aplicación de una combinación lineal de pesos a los atributos $a_1, \ldots, a_n$:

$$a_{n+1} = \sum_{i=1}^{n} a_i w_i \tag{26}$$

donde w_i es el peso que indica en qué grado contribuye el atributo a_i en la creación del nuevo atributo. En función de cómo se ajusten los pesos, el método y el resultado obtenido será uno u otro.

El método de análisis de componentes principales (*Principal Component Analysis*, PCA) es uno de los métodos más conocidos para la fusión y creación de nuevos atributos.

3.4.2. Métodos de reducción de casos

En minería de datos no siempre es posible disponer de todos los datos de la población que se estudia. Es en esta circunstancia en la que el concepto de "muestra" se convierte en relevante para el analista. A una muestra de una población se le exige que sus propiedades sean extrapolables a las propiedades de la población. Por ejemplo, cuando se hace una encuesta a una muestra poblacional se desea que esta sea lo más representativa de la población posible, sin sesgos evidentes.

Llamaremos **espacio muestral** al conjunto de muestras que se extrae de una población. La variable que asocia a cada muestra su probabilidad de extracción sigue lo que se llama **una distribución muestral**, cuyo estudio nos permitirá calcular la probabilidad que se tiene, dada una sola muestra, de acercarse a las propiedades de la población.

Respecto de las técnicas de muestreo, podemos clasificarlas en dos grupos: muestreo probabilístico y muestreo no probabilístico o subjetivo.

Técnicas de muestreo probabilístico

Se trata del conjunto de técnicas más aconsejable, puesto que permiten calcular la probabilidad de extracción de cualquiera de las muestras posibles.

A continuación enumeramos algunas de las más importantes:

- **Muestreo estratificado**: consiste en dividir la población de estudio en grupos homogéneos respecto de alguna de las características que se quiere medir. Una vez determinados los grupos, a estos se les aplicaría, por ejemplo, una técnica de muestreo sistemática. Casos de grupos pueden ser por sexo, por grupo de edad, por sector laboral, etc.

- **Muestreo sistemático**: si el número total de miembros de nuestra población es N y el número de miembros de la muestra que queremos tomar es n, entonces tendremos que el intervalo de selección será $\frac{N}{n}$. Escogemos al azar un miembro de la población q, y, a partir de este, se escogen los

siguientes miembros siguiendo el orden del intervalo de selección. Esto es $\{q + i\frac{N}{n} \mid i = 0, \ldots, n-1\}$.

- **Muestreo por estadios múltiples**: se divide la población en niveles estratificados, de modo que vamos tomando muestras de los distintos niveles considerados. Por ejemplo, si queremos realizar un estudio de intención de voto, empezaríamos por dividir la población total en regiones, localidades y barrios.

- **Muestreo por conglomerado**: esta técnica está indicada para poblaciones que ya de forma natural se encuentran divididas en grupos que contienen toda la variabilidad de la población total. Es decir, que el estudio de uno de los grupos es suficiente para extrapolar los resultados al resto de población.

Técnicas de muestreo no probabilístico o subjetivo

Se dan cuando no es posible determinar la probabilidad de extracción de una determinada muestra o incluso cuando ciertos elementos de la población no tienen posibilidades de ser seleccionados (individuos fuera de cobertura).

Como la selección de muestras no es aleatoria, tampoco aplica el concepto de estimación de errores de la muestra. Esta circunstancia provoca lo que se conoce como **sesgo de selección**, que es la distorsión provocada por el hecho de que la muestra tiene muchas limitaciones a la hora de representar toda la variabilidad de la población total.

A continuación enumeramos algunas de las más importantes:

- **Muestreo por cuotas**: podemos pensarlo como el muestreo por estratos, pero ponderado por cuotas. Es decir, se divide la población en estratos definidos por alguna variable de distribución conocida (por ejemplo, el sexo). Posteriormente, calculamos la proporción de cada estrato respecto de la población total. Finalmente, se multiplica cada proporción por el tamaño de cada muestra. Esta técnica es habitual en sondeos de opinión y estudios de mercado.

- **Muestreo de bola de nieve**: en esta técnica se selecciona inicialmente un pequeño grupo de individuos que nos ayudarán a encontrar más miembros representativos de la población. Está indicada para los casos en que la población se encuentra muy dispersa o incluso escondida.

4. Extracción y selección de atributos

En el proceso de descubrimiento de conocimiento en grandes volúmenes de datos es vital escoger las variables y características más adecuadas para presentar el algoritmo de minería de datos. Este problema puede tener diferentes enfoques, entre otros: escoger los mejores atributos de los datos a partir de su análisis preliminar, eliminar los atributos redundantes o que aportan poca información al problema que se desea resolver, o reducir la dimensionalidad de los datos generando nuevos atributos a partir de atributos existentes. En cualquiera de estos casos, el objetivo es reducir el coste espacial y computacional de creación del modelo, permitiendo además la creación de un modelo de similar o mejor calidad.

La **selección de atributos** consiste en escoger únicamente aquellos atributos que son realmente relevantes para el problema a resolver, descartando otros que no aportan información relevante para dicho objetivo. Por otra parte, la **extracción de atributos** trata de calcular nuevos atributos a partir de los existentes, de tal forma que los nuevos atributos resuman mejor la información que contienen, capturando la naturaleza de la estructura subyacente en los datos.

Existen métodos automáticos para la selección y extracción de atributos. No obstante, ambos métodos también pueden hacerse de forma manual usando conocimiento sobre el problema a resolver. Es importante subrayar que la selección o extracción manual de atributos requiere un experto en el dominio que analice y escoja los atributos más relevantes para el problema que se intenta resolver en cada caso. Este es un proceso *ad hoc* que requiere gran conocimiento del dominio del problema, así como de los datos que se utilizarán para el proceso de minería de datos.

Por ejemplo, el índice de masa corporal, que se define como el peso de una persona en kilogramos dividido por el cuadrado de su altura en metros, informa mejor del grado de obesidad de una persona que las dos variables originales por separado. Este tipo de conocimiento proviene del contexto o dominio del problema, y no debe ser descartado. No obstante, en la mayoría de casos será necesario recurrir a los métodos automáticos para extraer características de un conjunto de datos.

En este apartado nos centraremos en los métodos automáticos de selección y extracción de atributos.

4.1. Selección de atributos

Los métodos de selección de características o atributos (*feature selection*) permiten identificar los atributos que aportan información relevante para el proceso de minería de datos, o al revés, los atributos redundantes que no aportan información relevante a este proceso. En ambos casos, el objetivo es el mismo, elegir qué subconjunto de atributos es más beneficioso para resolver el problema en cuestión.

Dependiendo de si la selección de características usa o no información del método de clasificación posterior, podemos definir la siguiente taxonomía:

- Los algoritmos filtro (*filter*), donde los atributos o subconjuntos de atributos son evaluados de forma independiente del método de clasificación que se utilizará posteriormente.

- Los algoritmos empotrados (*wrappers*), donde el método de selección de características utiliza el clasificador que se usará posteriormente para evaluar qué característica o subconjunto de características es el más adecuado.

Los algoritmos empotrados, aunque suelen tener un buen rendimiento, pueden tener más tendencia a sobreaprender el conjunto de entrenamiento, perdiendo capacidad de generalización. Los algoritmos de selección de características han sido ampliamente estudiados, y se han desarrollado multitud de algoritmos, algunos específicos para determinados problemas (Guyon & Elisseeff, 2003).

Lectura complementaria

I. Guyon, A. Elisseeff. (2003). "An Introduction to Variable and Feature Selection". J. Mach. Learn. Res., Vol. 3:1157-1182. http://dl.acm.org/citation.cfm?id=944919.944968

En primer lugar, veremos brevemente algunos de los métodos de selección de atributos individuales, llamados **algoritmos univariantes**, más empleados para la selección de atributos:

- Selección de máxima relevancia (*Maximum relevance selection*). Utiliza el coeficiente de correlación entre cada característica y los resultados de clasificar un determinado conjunto de entrenamiento, obteniendo una lista ordenada de las características que mejor diferencian los datos.

- Selección basada en la información mutua. Mide la información mutua entre las variables aleatorias que modelan cada característica y las etiquetas de clasificación, escogiendo las características que maximizan esta información mutua.

- Métodos basados en test estadísticos. Aplicación de test estadísticos de hipótesis sobre los datos, como por ejemplo el *T-statistic* o el *chi-square*.

En segundo lugar, encontramos los métodos de selección de subconjuntos de atributos, llamados **algoritmos multivariantes**:

- Búsqueda exhaustiva (*Exhaustive search*). Consiste en definir un espacio de búsqueda y evaluar, mediante una función de coste, todas las posibles combinaciones de atributos. Solo es aplicable a problemas de dimensionalidad pequeña.

- Selección paso a paso (*StepWise selection*). Consiste en iterar un algoritmo en el cual a cada paso o bien se añade al conjunto de atributos seleccionados aquel atributo que aumenta el rendimiento global del conjunto, o bien se elimina aquel atributo que hace que el rendimiento del subconjunto empeore.

- Ramificación y poda (*Branch and bound*). Consiste en aplicar la técnica de búsqueda de *branch and bound* en el espacio de las posibles combinaciones de características. Esta técnica reduce de forma muy notable la búsqueda exhaustiva de la solución.

4.2. Extracción de atributos

El objetivo de la extracción de características es obtener un espacio de dimensionalidad inferior, que preserve al máximo posible los datos útiles y elimine la información redundante. A diferencia de la selección de atributos, en la extracción de atributos se pueden crear nuevos atributos a partir de los existentes en el conjunto de datos inicial.

Las técnicas que se describen en este subapartado son conocidas como **técnicas de factorización matricial**, puesto que descomponen una matriz de datos como el producto de matrices más simples. En cualquier caso, la factorización de una matriz no es única, y cada técnica pone de manifiesto aspectos diferentes de la información contenida en los datos originales.

4.2.1. Análisis de componentes principales (PCA)

El análisis de componentes principales (*Principal Component Analysis*, PCA) nos puede ayudar a solucionar problemas de reducción de dimensionalidad y extracción de características en nuestros datos de forma automática. El PCA es un algoritmo muy conocido y usado en el análisis de datos, y tiene muchas posibles aplicaciones diferentes. Informalmente, se puede definir como la técnica que intenta conseguir una representación de un conjunto de datos en un espacio de dimensionalidad más reducida, minimizando el error cuadrático cometido.

El algoritmo PCA reduce la dimensionalidad a través de una transformación lineal que escoge un nuevo sistema de coordenadas para el conjunto original de datos (es decir, realiza una rotación del espacio d-dimensional), en el

cual la varianza de mayor tamaño del conjunto de datos se recoge en el primer eje (llamado el primer componente principal), la segunda varianza más grande en el segundo eje, y así sucesivamente. Si los datos presentan alguna "inclinación" (en sentido geométrico), los ejes calculados por el PCA siguen dicha inclinación para capturar la misma varianza con un menor volumen del hiperparalelepípedo que contiene todos los datos.

Este algoritmo se basa en la matriz de covarianzas o correlaciones de los datos originales, de forma que calcula los vectores propios de esta matriz y se aplica a los datos originales para conseguir la transformación lineal. Generalmente se utiliza la matriz de correlación cuando los datos no son dimensionalmente homogéneos o el orden de magnitud de las variables aleatorias medidas no es el mismo. La matriz de covarianzas se usará cuando los datos son dimensionalmente homogéneos y presentan valores medios similares.

Podemos expresar el resultado del proceso de extracción de características como el producto matricial:

$$S_{q,n} = P_{q,m} \cdot D^T_{m,n} \tag{27}$$

donde D es el conjunto de datos original, S es la matriz de los resultados (donde $q \ll m$), y P es la matriz de proyección, que permite realizar la extracción de características. Es importante notar que la matriz de datos se presenta de forma traspuesta a la forma en que la hemos definido en este texto, es decir, las filas representan los atributos y las columnas las instancias.

El objetivo consiste en encontrar la matriz P que reduce la dimensionalidad de los datos, minimizando el error cuadrático cometido en este proceso de reducción de la dimensionalidad.

El proceso para el cálculo de la matriz de soluciones D es el siguiente:

1) Calcular la media de los datos de cada columna de la matriz D, para obtener un conjunto de datos centrados en su origen $\hat{D}$.

2) Calcular la matriz de covarianza C de los datos como:

$$C = \frac{1}{N}\hat{D} \times \hat{D}^T \tag{28}$$

3) Coger como matriz P, los q primeros vectores propios con mayor valor propio asociado de la matriz de covarianza C.

Aplicando el PCA, obtenemos una serie de vectores $S = \{s_1, \ldots, s_q, \ldots, s_m\}$ donde m es el número de atributos. En un problema de reducción de dimensio-

nalidad nos quedaríamos con los q primeros vectores que explicasen la mayor parte de la varianza, y que escogemos en el tercer paso del algoritmo detallado anteriormente. Los coeficientes de los vectores asociados a cada componente marcan la importancia de cada uno de los nuevos atributos, de forma que los valores altos serán más importantes, mientras que podemos prescindir del resto.

4.2.2. Descomposición en valores singulares (SVD)

El método de la descomposición en valores singulares (*Singular Value Decomposition*, SVD) es una herramienta del álgebra lineal que permite descomponer una matriz de datos expresándola como la suma ponderada de sus vectores propios.

El resultado de la descomposición factorial de la matriz D de tamaño $m \times n$ se representa mediante:

$$D_{m,n} = U_{m,m} \cdot S_{m,n} \cdot V^T_{n,n} \tag{29}$$

donde S es una matriz diagonal de tamaño $m \times n$ con componentes no negativos y U, V son matrices unitarias de tamaño $m \times m$ y $n \times n$ respectivamente.

A los componentes diagonales no nulos de la matriz S se les conoce como valores singulares, mientras que las m columnas de la matriz U y las n columnas de la matriz V se denominan vectores singulares por la izquierda y por la derecha, respectivamente. El número máximo de valores singulares diferentes de la matriz D está limitado por el rango máximo de dicha matriz, $r = min\{m,n\}$.

Aplicando este procedimiento a una matriz de datos, obtendremos una representación de la información en función de unos pocos vectores singulares, y por lo tanto dispondremos de una representación de los datos en un espacio de dimensionalidad reducida.

4.2.3. Factorización de matrices no negativas (NNMF)

El método de factorización de matrices no negativas (*Non-Negative Matrix Factorization*, NMF o NNMF) realiza una descomposición en factores de una matriz de datos no negativa D de la siguiente forma:

$$D_{n,m} \approx W_{n,q} \cdot F_{q,m} \tag{30}$$

donde las matrices no negativas F y W son conocidas como matriz de características y matriz de pesos, respectivamente.

La matriz de características F, de tamaño $q \times m$, define q características y pondera cada una de ellas mediante la importancia que tiene cada uno de los m atributos. Las características corresponden a temas genéricos que han sido definidos a partir de la agrupación de atributos en diferentes instancias. La componente i,j de F representa la importancia del atributo j en la característica i.

Por otra parte, la matriz de pesos W le atribuye un peso a cada una de las características en función de su importancia en cada una de las n instancias. Sus q columnas corresponden a las características y sus n filas a las instancias del conjunto de datos analizados, de forma que la componente i,j de W indica la importancia que tiene la característica j en la instancia i.

Uno de los parámetros que deben definirse para aplicar NNMF es el número de características q que se desean. Un valor demasiado alto resulta en un etiquetado demasiado específico de las instancias, mientras que un valor excesivamente pequeño agrupa las instancias en unas pocas características demasiado genéricas y, por tanto, carentes de información relevante. La matriz de características F es equivalente a la matriz de vectores propios que obteníamos con otras técnicas de factorización de datos, como por ejemplo PCA. La diferencia principal entre NNMF y PCA radica en que en NNMF la matriz F es no negativa y, por tanto, la factorización de D puede interpretarse como una descomposición acumulativa de los datos originales.

Conviene precisar que la descomposición NNMF no es única, es decir, dado un número de características q hay más de una forma de descomponer los datos en la forma de la ecuación 30. Lo ideal es aplicar el procedimiento e interpretar los resultados para obtener información adicional de los datos. El algoritmo que se encarga de realizar esta factorización no es trivial y escapa al alcance de este módulo didáctico.

5. Conjuntos desbalanceados de datos

Los datos desbalanceados se refieren a una situación en la que las clases o categorías objetivo de un conjunto de datos no están representadas de manera equitativa. Es decir, algunas clases tienen un número significativamente mayor de ejemplos que otras. Esto implica diversos problemas y desafíos durante el entrenamiento y la evaluación de los modelos de aprendizaje automático, dado que los modelos tienen tendencia a clasificar mejor las clases más populares, dejando de prestar atención a las clases minoritarias. Por ejemplo, en el caso de las transacciones bancarias, un porcentaje altísimo son legales, mientras que unas pocas (es típico que sea solamente el 0,1 %) son fraudulentas. Un clasificador que siempre prediga que una transacción es legal acertará el 99,9 % de las veces, pero no servirá de nada para detectar el 0,1 % residual de transacciones fraudulentas.

Formalmente, supongamos que tenemos un conjunto de datos con un total de n elementos y C clases diferentes. Denotemos como N_i el número de elementos de la clase i, donde i toma valores desde 1 hasta C. La distribución desigual de los datos se puede expresar mediante la siguiente desigualdad:

$$N_1 \ll N_2 \ll ... \ll N_i \ll \ldots \ll N_C$$

donde $\ll$ representa una diferencia significativa entre el número de elementos de las diferentes clases.

Los problemas derivados de trabajar con conjuntos de datos desbalanceados son múltiples:

1) Sesgo del modelo: cuando las clases están desbalanceadas, el modelo tiende a inclinarse hacia las clases mayoritarias debido a que tienen más elementos para aprender de ellos. Como resultado, el modelo puede tener dificultades para reconocer y clasificar correctamente las clases minoritarias.

2) Baja precisión y sensibilidad: en presencia de datos desbalanceados, la precisión y la sensibilidad del modelo pueden verse afectadas. La precisión mide la proporción de elementos clasificados correctamente en una clase específica, mientras que la sensibilidad mide la proporción de elementos de esa clase que fueron identificados correctamente. Es común que un modelo tenga una alta precisión para la clase mayoritaria pero una baja sensibilidad para las clases minoritarias.

3) Sobreajuste en clases minoritarias: debido a que hay menos elementos disponibles para las clases minoritarias, el modelo puede sobreajustarse (*overfitting*) a estos elementos durante el entrenamiento, lo que puede afectar negativamente a su capacidad de generalización en datos nuevos.

4) Evaluación engañosa: las métricas de rendimiento, como el error de clasificación o la exactitud (*accuracy*), pueden no ser representativas del rendimiento real del modelo en situaciones desbalanceadas. Incluso un modelo trivial que clasificara todos los elementos como la clase mayoritaria podría obtener una alta exactitud, tanta como el porcentaje de dicha clase mayoritaria respecto al total de elementos del conjunto de datos.

Para abordar el problema de datos desbalanceados, se pueden aplicar diversas técnicas de manejo de desequilibrio de clases, como el sobremuestreo (*oversampling*) de las clases minoritarias, el submuestreo (*undersampling*) de las clases mayoritarias, la generación de datos sintéticos y el uso de métricas de rendimiento adecuadas. Se trata de un problema muy importante que no puede ser ignorado y que debe ser, como mínimo, explorado.

5.1. Técnicas de sobremuestreo

El sobremuestreo (*oversampling*, en inglés) es una técnica que busca aumentar el número de elementos de las clases minoritarias para equilibrar la distribución de clases y mejorar el rendimiento del modelo cuando se trabaja con conjuntos de datos desbalanceados. El objetivo, en cualquiera de las diferentes técnicas que veremos a continuación, siempre es generar nuevas muestras de las clases minoritarias para aliviar el problema de la baja representación de estas clases en el conjunto de datos.

A continuación, se describen algunas de las principales técnicas de sobremuestreo:

- **Duplicación** (*Duplicate*): esta técnica consiste en duplicar los elementos de las clases minoritarias hasta alcanzar un número similar de elementos que las clases mayoritarias. El proceso es muy simple, en primer lugar, se seleccionan las clases minoritarias con menos elementos y, a continuación, se duplican aleatoriamente elementos de estas clases hasta igualar el número de elementos de la clase mayoritaria.

- **SMOTE** (*Synthetic Minority Over-sampling Technique*): SMOTE es una técnica que genera elementos sintéticos de las clases minoritarias a partir de los elementos existentes. Crea muestras sintéticas mediante la interpolación entre elementos de una clase minoritaria y sus vecinos más cercanos en el espacio de características. El proceso, en términos generales, de esta técnica consiste en:

1) Seleccionar una instancia de la clase minoritaria.

2) Elegir los *k* vecinos más cercanos de la misma clase (generalmente se suele emplear una *k*=5).

3) Generar un nuevo elemento sintético como una combinación lineal de la instancia seleccionada y uno de sus *k* vecinos cercanos.

- **ADASYN** (*Adaptive Synthetic Sampling*): ADASYN es una extensión de SMOTE que ajusta la densidad de los elementos sintéticos generados en función de la dificultad de la clasificación de cada elemento. Es decir, esta técnica se centra en las regiones de la clase minoritaria que son especialmente difíciles de clasificar. El proceso, en términos generales, de esta técnica consiste en:

  1) Seleccionar una instancia de la clase minoritaria.

  2) Calcular el número de elementos sintéticos que se deben generar para esta instancia en función de la densidad de sus vecinos de otras clases.

  3) Generar elementos sintéticos cercanos a la instancia seleccionada.

Estas técnicas son solo algunas de las más utilizadas para lidiar con problemas de conjuntos de datos desbalanceados. Cada enfoque tiene sus ventajas y desventajas, y la elección del método dependerá del conjunto de datos específico y del problema en cuestión.

El sobremuestreo debe aplicarse con precaución, ya que un aumento excesivo de elementos sintéticos puede llevar a un sobreajuste del modelo. Por lo tanto, es esencial evaluar el rendimiento del modelo en conjuntos de validación o prueba independientes para asegurarse de que no se esté introduciendo un sesgo.

5.2. Técnicas de submuestreo

El submuestreo (*undersampling*, en inglés) es una técnica que busca reducir el número de elementos de las clases mayoritarias para equilibrar la distribución de clases y mejorar el rendimiento del modelo en conjuntos de datos desbalanceados. El objetivo, en cualquiera de las diferentes técnicas que veremos a continuación, siempre es eliminar o descartar un subconjunto de muestras de las clases mayoritarias para aliviar el problema de la excesiva representación de estas clases en el conjunto de datos.

A continuación, se describen algunas de las principales técnicas de submuestreo:

- **Eliminación aleatoria** (Random Undersampling): esta técnica consiste en eliminar aleatoriamente elementos de las clases mayoritarias hasta que al-

cancen un número similar de elementos que las clases minoritarias. En términos generales, el proceso de esta técnica consiste en:

1) Seleccionar las clases mayoritarias con más elementos.

2) Eliminar aleatoriamente elementos de estas clases hasta igualar el número de elementos de la clase minoritaria.

- **Tomek Links**: los enlaces de Tomek son pares de ejemplos (uno de la clase minoritaria y otro de la clase mayoritaria) que son muy cercanos entre sí, pero que pertenecen a clases diferentes. Esta técnica consiste en eliminar los elementos de la clase mayoritaria para mejorar la separabilidad entre las clases y dar relevancia a las clases con baja representación en el conjunto de datos. El proceso consiste en:

  1) Encontrar los enlaces de Tomek, que son pares de elementos cercanos y de clases diferentes.

  2) Eliminar los elementos de la clase mayoritaria en estos enlaces.

- **Edited Nearest Neighbors**: esta técnica utiliza un clasificador para identificar elementos de la clase mayoritaria que han sido clasificados de forma errónea. La base de esta técnica es eliminar los elementos de la clase mayoritaria que son clasificados incorrectamente, reduciendo así la presencia de elementos ruidosos para mejorar la clasificación de las clases minoritarias. El proceso consiste en:

  1) Entrenar un clasificador con el conjunto de datos original.

  2) Identificar los elementos de la clase mayoritaria que son clasificados incorrectamente por el clasificador.

  3) Eliminar los elementos identificados de la clase mayoritaria.

Estos métodos son solo algunas de las técnicas comunes de submuestreo utilizadas para tratar problemas de conjuntos de datos desbalanceados. Al igual que con el caso anterior, la elección del método de submuestreo dependerá del conjunto de datos específico y del problema en cuestión.

En general, se considera que el submuestreo es útil cuando el número de elementos en el conjunto de datos es muy grande. Sin embargo, estas técnicas pueden implicar la pérdida de información importante, especialmente si los elementos eliminados contienen información relevante para la clasificación. Por lo tanto, es importante evaluar cuidadosamente el rendimiento del modelo después de aplicar técnicas de submuestreo y considerar otras alternativas si es necesario.

5.3. Métricas de evaluación

Como ya hemos comentado anteriormente, cuando se trata con conjuntos de datos desbalanceados, las métricas de evaluación convencionales como la exactitud (*accuracy*) pueden ser engañosas y no reflejar adecuadamente el rendimiento del modelo, en especial si hay una clase mayoritaria claramente dominante. Por lo tanto, es fundamental escoger las métricas más apropiadas para reportar los resultados de los modelos de aprendizaje automático.

En particular, se recomienda el uso de las siguientes métricas de evaluación para lidiar con conjuntos de datos desbalanceados:

- Precisión (*Precision*): la precisión mide la proporción de elementos clasificados correctamente como positivos (verdaderos positivos, TP) entre todos los elementos clasificados como positivos (verdaderos positivos, TP, más falsos positivos, FP). Es importante aplicar esta métrica para las clases minoritarias para evaluar la bondad del modelo.

- Sensibilidad (*Recall* o *Sensitivity*): la sensibilidad o *recall*, también conocida como "exhaustividad", mide la proporción de elementos positivos clasificados correctamente (verdaderos positivos, TP) entre todos los elementos que realmente son positivos (verdaderos positivos, TP, más falsos negativos, FN).

- F1-Score: el F1-Score es una métrica que combina precisión y sensibilidad, proporcionando un equilibrio entre ambas.

- Área bajo la curva ROC (ROC-AUC): la curva ROC (*Receiver Operating Characteristic*) muestra la tasa de verdaderos positivos frente a la tasa de falsos positivos para diferentes umbrales de clasificación. El área bajo la curva ROC (ROC-AUC) es una métrica que resume el rendimiento del modelo considerando todos los posibles umbrales de clasificación.

Estas métricas se pueden aplicar cuando lidiamos con conjuntos de datos desbalanceados, ya que tienen en cuenta tanto los verdaderos positivos como los falsos positivos y falsos negativos. Es decir, son especialmente útiles cuando hay clases minoritarias que son importantes pero tienen pocos ejemplos en el conjunto de datos.

Además, es recomendable utilizar técnicas de validación cruzada estratificada para asegurarse de que los conjuntos de entrenamiento y prueba mantengan la misma proporción de clases, evitando así problemas adicionales con datos desbalanceados durante la evaluación del modelo.

Resumen

En este módulo didáctico hemos introducido los principales conceptos relacionados con el preprocesamiento de datos en entornos de aprendizaje automático.

En concreto, hemos iniciado el texto con un repaso de conceptos estadísticos preliminares que son importantes para poder comprender la naturaleza de los datos. A continuación, nos hemos centrado en las tareas básicas de preparación de los datos, que incluyen la limpieza de los datos, la normalización y estandarización de los datos, la discretización y reducción de la dimensionalidad.

Seguidamente, nos hemos centrado en los procesos de extracción y selección de atributos, que permiten reducir el número de características de un conjunto de datos para hacer el problema más tratable.

Finalmente, en el último bloque de este módulo didáctico, hemos tratado la problemática de los conjuntos de datos desbalanceados, introduciendo las técnicas básicas para aumentar el número de instancias de las clases minoritarias o, por el contrario, reducir el número de instancias de las clases mayoritarias.

Glosario

conjunto de datos desbalanceados *m* Situación que se produce cuando en un conjunto de datos con elementos de diferentes clases o categorías existe un número mucho mayor o mucho menor de elementos de algunas categorías con respecto al resto. Esto puede causar que los modelos construidos tengan tendencia a ignorar las categorías minoritarias.

entropía *f* La entropía de una variable discreta mide el grado de desorden presente entre los posibles valores de dicha variable. A mayor entropía, mayor esfuerzo se deberá hacer para realizar una predicción al respecto.

estandarización de datos *f* Se trata de realizar una transformación en cada una de las variables del conjunto original de forma que la variable resultante tenga media 0 y varianza 1.

normalización de datos *f* Se trata de realizar una transformación en cada una de las variables del conjunto original para que todas ellas estén en un rango concreto, normalmente [0,1]. De esta manera se evita que una variable con un rango de valores muy elevado predomine sobre el resto.

overfitting *m* Vease *sobreajuste*

sobreajuste *m* El sobreajuste se produce cuando el algoritmo usado para crear un modelo captura en exceso la naturaleza intrínseca del conjunto de datos usado en el proceso de creación, imposibilitando que el modelo generalice bien para datos nunca observados con anterioridad, creando un sesgo.
en overfitting

Bibliografía

Cichosz, Pawel (2015). *Data mining algorithms. Explained using R*. Wiley.

Géron, Aurélien (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition*. O'Reilly Media, Inc.

Gironés Roig, J.; Casas Roma, J.; Minguillón Alfonso, J.; Caihuelas Quiles, R. (2017). *Minería de datos: Modelos y algoritmos*. Barcelona: Editorial UOC.

Jonge, Edwin de; Loo, Mark van der (2013). *An introduction to data cleaning with R*. The Hague [u. a.]: Statistics Netherlands.