3_Preprocesamiento_Teoría

July 6, 2025

1 Pasos que realizar para el preprocesamiento:

1.1 Limpieza del dataset

La mayor parte de los algoritmos de aprendizaje automático tiene problemas o no funcionan bien cuando tenemos columnas con datos nulos. Es decir si algunas filas tiene datos y otras no.

Si fuera así deberíamos de seguir los siguientes pasos:

- Eliminar los datos con los valores nulos, reduciendo así el número de datos.
- Eliminar por completo, la columna que contiene valores nulos, reduciendo el número de columnas del dataset.
- Asignar un valor al campo de la columna con valores nulos, por ejemplo el valor medio (mediana).

1.2 Variables categóricas (Dummies Variables)

En general, los algoritmos de machine learning no trabajan bien con datos con valores que no sean numéricos. Es por ello, que aquellos datos con valores que contengan un texto o sean una categoría, debemos transformarlos a valores numéricos.

Ejemplo de columnas no numéricas: «Color» que nos indica el color en inglés (Red, Blue, etc) y «Spectral_Class» que es otro dato categórico (M, O, A, etc).

Tipos de datos categóricos

Los datos categóricos se pueden clasificar en términos generales en dos tipos:

- **Datos nominales:** este tipo de datos representan categorías sin ningún orden inherente. Los ejemplos incluyen género (masculino, femenino), color (rojo, azul, verde) y país (EE. UU., India, Reino Unido).
- **Datos ordinales:** este tipo de datos representa categorías con un orden o clasificación significativo. Los ejemplos incluyen el nivel educativo (escuela secundaria, licenciatura, maestría, doctorado) y la satisfacción del cliente (baja, media, alta).

Técnicas de codificación Exploremos las técnicas más utilizadas:

1. Codificación de etiquetas (Label encoding)

La codificación de etiquetas es un método simple y directo que asigna un número entero único a cada categoría. Este método es adecuado para datos ordinales donde el orden de las categorías es significativo. Caso de Uso: Aplicable para casos en los que el ordenamiento de categorías tiene relevancia analítica.

```
data = ['red', 'blue', 'green', 'blue', 'red']

0 => 'blue'
1 => 'green'
2 => 'red'

data = [2 0 1 0 2]
```

2. Codificación en caliente (One-hot Encoding)

One-Hot Encoding convierte datos categóricos en una matriz binaria, donde cada categoría está representada por un vector binario. Este método es adecuado para datos nominales. Caso de uso: Más apropiado para aquellas situaciones en las que las categorías no tienen un orden inherente o existe una distinción clara entre ellas.

red	blue	green
0	1	0
1	0	0
0	0	1

```
data = ['red', 'blue', 'green', 'blue', 'red']

data = [[0. 1. 0.]
        [1. 0. 0.]
        [0. 0. 1.]
        [1. 0. 0.]
        [0. 1. 0.]]
```

Ventajas y desventajas de cada técnica de codificación

Técnica de codificación	Ventajas	Desventajas
Codificación de etiquetas	- Simple y fácil de implementar - Adecuado para datos ordinales	- Introduce relaciones ordinales arbitrarias para datos nominales- Puede no funcionar bien con valores atípicos
Codificación en caliente	- Adecuado para datos nominales - Evita introducir relaciones ordinales- Mantiene información sobre los valores de cada variable	- Puede provocar un aumento de la dimensionalidad y la escasez. - Puede provocar un sobreajuste, especialmente con muchas categorías y tamaños de muestra pequeños.

1.3 Reducción de datos

La reducción de datos es una técnica utilizada en la minería de datos para reducir el tamaño de un conjunto de datos y al mismo tiempo preservar la información más importante. Esto puede resultar beneficioso en situaciones en las que el conjunto de datos es demasiado grande para procesarlo de forma eficiente o en las que el conjunto de datos contiene una gran cantidad de información irrelevante o redundante.

Existen varias técnicas diferentes de reducción de datos que se pueden utilizar en la minería de datos, que incluyen:

- **Muestreo de datos:** esta técnica implica seleccionar un subconjunto de datos con los que trabajar, en lugar de utilizar todo el conjunto de datos. Esto puede resultar útil para reducir el tamaño de un conjunto de datos y al mismo tiempo preservar las tendencias y patrones generales de los datos.
- **Reducción de dimensionalidad:** esta técnica implica reducir la cantidad de funciones en el conjunto de datos, ya sea eliminando funciones que no son relevantes o combinando varias funciones en una sola.
- **Compresión de datos:** esta técnica implica el uso de técnicas como la compresión con o sin pérdidas para reducir el tamaño de un conjunto de datos.
- **Discretización de datos:** esta técnica implica convertir datos continuos en datos discretos dividiendo el rango de valores posibles en intervalos o contenedores.
- **Selección de características:** esta técnica implica seleccionar un subconjunto de características del conjunto de datos que sean más relevantes para la tarea en cuestión.

Es importante tener en cuenta que la reducción de datos puede tener un equilibrio entre la precisión y el tamaño de los datos. Cuantos más datos se reduzcan, menos preciso será el modelo y menos generalizable.

En conclusión, la reducción de datos es un paso importante en la minería de datos, ya que puede ayudar a mejorar la eficiencia y el rendimiento de los algoritmos de aprendizaje automático al reducir el tamaño del conjunto de datos. Sin embargo, es importante ser consciente del equilibrio entre el tamaño y la precisión de los datos y evaluar cuidadosamente los riesgos y beneficios antes de implementarlos.

Ventajas o desventajas de la Reducción de Datos en Minería de Datos:

La reducción de datos en la minería de datos puede tener una serie de ventajas y desventajas.

Ventajas: - **Eficiencia mejorada:** la reducción de datos puede ayudar a mejorar la eficiencia de los algoritmos de aprendizaje automático al reducir el tamaño del conjunto de datos. Esto puede hacer que trabajar con grandes conjuntos de datos sea más rápido y práctico.

- **Rendimiento mejorado:** la reducción de datos puede ayudar a mejorar el rendimiento de los algoritmos de aprendizaje automático al eliminar información irrelevante o redundante del conjunto de datos. Esto puede ayudar a que el modelo sea más preciso y robusto.
- **Costos de almacenamiento reducidos:** la reducción de datos puede ayudar a reducir los costos de almacenamiento asociados con grandes conjuntos de datos al reducir el tamaño de los datos.
- **Interpretabilidad mejorada:** la reducción de datos puede ayudar a mejorar la interpretabilidad de los resultados al eliminar información irrelevante o redundante del conjunto de datos.

Desventajas: - **Pérdida de información:** la reducción de datos puede resultar en una pérdida de información si se eliminan datos importantes durante el proceso de reducción.

- **Impacto en la precisión:** la reducción de datos puede afectar la precisión de un modelo, ya que reducir el tamaño del conjunto de datos también puede eliminar información importante que se necesita para realizar predicciones precisas.
- **Impacto en la interpretabilidad:** la reducción de datos puede dificultar la interpretación de los resultados, ya que eliminar información irrelevante o redundante también puede eliminar el contexto necesario para comprender los resultados.
- **Costos computacionales adicionales:** la reducción de datos puede agregar costos computacionales adicionales al proceso de minería de datos, ya que requiere tiempo de procesamiento adicional para reducir los datos.

En conclusión, la reducción de datos puede tener ventajas y desventajas. Puede mejorar la eficiencia y el rendimiento de los algoritmos de aprendizaje automático al reducir el tamaño del conjunto de datos. Sin embargo, también puede provocar una pérdida de información y dificultar la interpretación de los resultados. Es importante sopesar los pros y los contras de la reducción de datos y evaluar cuidadosamente los riesgos y beneficios antes de implementarla.

Las técnicas empleadas para la reducción de los datos son PCA y LDA.

Análisis de Componentes Principales (PCA)

El análisis de componentes principales (PCA) es una técnica de reducción de dimensionalidad lineal que se puede utilizar para extraer información de un espacio de alta dimensión proyectándola en un subespacio de menor dimensión. Intenta preservar las partes esenciales que tienen más variación de los datos y eliminar las partes no esenciales con menos variación.

Una cosa importante a tener en cuenta sobre PCA es que es una técnica de reducción de dimensionalidad no supervisada , puede agrupar los puntos de datos similares en función de la correlación de características entre ellos sin supervisión (o etiquetas).

es un procedimiento estadístico que utiliza una transformación ortogonal para convertir un conjunto de observaciones de variables posiblemente correlacionadas (entidades cada una de las cuales toma varios valores numéricos) en un conjunto de valores de variables linealmente no correlacionadas llamadas componentes principales.

Pero, ¿dónde se puede aplicar PCA?

Visualización de datos: cuando se trabaja en cualquier problema relacionado con datos, el desafío en el mundo actual es el gran volumen de datos y las variables/características que definen esos datos. Para resolver un problema donde los datos son la clave, necesita una amplia exploración de datos, como descubrir cómo se correlacionan las variables o comprender la distribución de algunas variables. Teniendo en cuenta que hay una gran cantidad de variables o dimensiones a lo largo de las cuales se distribuyen los datos, la visualización puede ser un desafío y casi imposible.

Por lo tanto, PCA permite visualizar los datos en un espacio 2D o 3D a simple vista.

Aceleración del algoritmo de aprendizaje automático (ML): dado que la idea principal de PCA es la reducción de la dimensionalidad, puede aprovecharla para acelerar el entrenamiento y el tiempo de prueba de su algoritmo de aprendizaje automático, teniendo en cuenta que sus datos tienen muchas características y que el aprendizaje del algoritmo ML es demasiado lento.

En un nivel abstracto, toma un conjunto de datos que tiene muchas características y simplifica ese conjunto de datos seleccionando algunas Principal Componentes de las características originales.

¿Qué es un componente principal?

Los componentes principales son la clave de PCA. En términos sencillos, cuando los datos se proyectan en una dimensión más baja (suponga tres dimensiones) desde un espacio más alto, las tres dimensiones no son más que los tres componentes principales que capturan (o contienen) la mayor parte de la variación (información) de sus datos .

Los componentes principales tienen dirección y magnitud. La dirección representa a través de qué ejes principales se distribuyen principalmente los datos o tienen la mayor variación y la magnitud indica la cantidad de variación que el Componente principal captura de los datos cuando se proyecta en ese eje. Los componentes principales son una línea recta y el primer componente principal tiene la mayor variación en los datos. Cada componente principal posterior es ortogonal al último y tiene una varianza menor. De esta forma, dado un conjunto de "x" variables correlacionadas sobre "y" muestras, se obtiene un conjunto de "u" componentes principales no correlacionados sobre las mismas "y" muestras.

La razón por la que obtiene componentes principales no correlacionados de las características originales es que las características correlacionadas contribuyen al mismo componente principal, reduciendo así las características de datos originales a componentes principales no correlacionados; cada uno representa un conjunto diferente de características correlacionadas con diferentes cantidades de variación.

Cada componente principal representa un porcentaje de la variación total capturada de los datos.

1.4 Escalado de los datos

Este será una de las principales tareas que realicemos en el preprocesamiento de datasets. Los principales algoritmos de machine learning que existen no funcionan muy bien cuando existen una gran diferencia entre los valores de una columna. Si tenemos una columna «L» que contiene valores muy distante, por ejemplo, tiene un valor mínimo de 0 y máximo de 849820. Esto es algo que debemos evitar, ya que los algoritmos de aprendizaje no funcionan bien en estos casos.

Existen dos formas claras de solucionar estos problemas de escalas: la normalización de valores y la estandarización.

Normalización

El escalado **min-max** o normalización, es una técnica común a la hora de solucionar el problema de tener diferentes escalas en los valores de una columna. El objetivo que se consigue con esta técnica es que todos los valores de una columna estén comprendido en el intervalo [0-1]. De forma matemática, lo que estamos haciendo es a cada valor le restamos el mínimo y lo dividimos entre el valor máximo.

Otras maneras de utilizar la normalización, además de min-max son:

- **Normalización min-max:** se calcula de la siguiente fórmula

$$x_s = \frac{x - x_{min}}{x_{max} - x_{min}} * (max - min) + min$$

  donde:

- x es la variable en su escala original
- xmax y xmin son el máximo y el mínimo

- *max* y *min* son el máximo y el mínimo pre-definidos (es decir que queremos obtener tras el escalamiento)
- xs es la variable obtenida tras el escalamiento

¿Cuándo NO usar MinMaxScaler?

- Cuando la distribución tiene un sesgo Una distribución con sesgo es cuando su forma se aleja demasiado de una distribución normal o gaussiana (campana simétrica).

En este caso NO se recomienda el uso de MinMaxScaler pues al escalar los datos comprimimos la distribución de los datos a un rango más pequeño que el original. Es decir que desaprovechamos todo el rango de valores disponible tras el escalamiento

- Cuando los datos tienen valores extremos (outliers) Los valores extremos (u outliers) son simplemente datos cuyos valores se encuentran excepcionalmente fuera del rango de valores de la mayoría de nuestros datos.

En este caso tampoco se recomienda el uso de minmaxscaler. Y esto se debe a que por tratarse de valores extremos, estos outliers generalmente corresponderán a los valores xmax y xmin que aparecen en la ecuación de escalamiento de nuestra variable original.

Así que al hacer el escalamiento estos valores extremos quedarán mapeados al máximo y mínimo en el rango resultante, mientras que los valores no extremos (que es la mayoría de los datos) quedarán mapeados dentro de un rango resultante muchísimo menor.

Es decir que si usamos minmaxscaler cuando tenemos valores extremos en nuestros datos hace que la mayoría de nuestros datos (que no son extremos) queden como resultado comprimidos a un rango de valores muy pequeño.

¿Cuándo podemos usar MinMaxScaler? Teniendo en cuenta lo anterior, se sugiere el uso de MinMaxScaler en estas situaciones:

1. Cuando tenemos claro el rango de valores esperado a la salida del escalamiento
2. Cuando la distribución de los datos NO tiene demasiado sesgo
3. Cuando los datos NO contienen outliers

Estandarización

La otra forma de realizar el escalado de valores que vamos a ver se denomina estandarización. Matemáticamente hablando, lo que estamos haciendo en este proceso es restar la media de los valores y dividir por la desviación estándar de los mismos. De esta forma los valores obtenidos tendrán una media de cero y una varianza de uno.

Existen algunas diferencias notables con respecto a la normalización. En primer lugar los valores obtenidos por la estandarización no están acotados en ningún rango ([0-1] por ejemplo).

En segundo lugar, este método consigue solucionar el problema de valores atípicos u outliers que presenta la normalización. Por ejemplo, supongamos que un atributo de temperatura contiene valores entre 0 y 100 por regla general. Por un error de medición, tenemos un valor de 10000 que se consideraría un outlier. Si aplicamos la normalización, la mayor parte de valores estarían en el rango 0-0.1. Sin embargo, esto no se da con la estandarización.

En sklearn tenemos el método StandardScaler() para calcularlo usamos los siguiente:

- **Normalización Z-score:** sigue la siguiente fórmula:

$$z = (x - u)/s$$

donde:

- x el valor de la muestra
- u es la media de los datos
- s es la desviación estandar de los datos.

Otras maneras de utilizar la normalización, además de min-max son:

- Normalización Z-score
- Normalizado por escala decimal

Así pues podemos encontrarnos con distintas etapas del preprocesamiento:

- **Data cleaning:** la limpieza de datos elimina ruido y resuelve las inconsistencias en los datos.
- **Data integration:** con la Integración de datos se migran datos de varias fuentes a una fuente coherente como un Data Warehouse.
- **Data transformation:** la transformación de datos sirve para normalizar datos de cualquier tipo.
- **Data reduction:** la reducción de datos reduce el tamaño de los datos agregandolos.

1.5 ETL - Extract, Transform, Load

Las herramientas ETL, ya poseen la mayoría de las técnicas de procesamiento de datos mencionadas anteriormente como la migración de datos y la transformación de datos, esto hace que el seguimiento de estas prácticas de limpieza de datos resulte mucho más conveniente. Además, tales herramientas ETL permiten a los usuarios especificar los tipos de transformaciones que desean realizar con sus datos.

1.5.1 LIBRERÍA SCIKIT-LEARN (sklearn)

Para realizar este paso podemos ir a la librería sklearn para ver los distintos tipos disponibles: https://scikit-learn.org/stable/modules/preprocessing.html