ITESO, Universidad Jesuita de Guadalajara

Departamento de Matemáticas y Física

Ingeniería de características

Escalamiento de variables

Dr. Gaddiel Desirena López

Primavera 2024

Muchos algoritmos de aprendizaje automático son sensibles a la escala y magnitud de las características. En particular, los coeficientes de los modelos lineales dependen de la escala, es decir, cambiar la escala de la característica cambiará el valor de los coeficientes. En los modelos lineales, así como en los algoritmos que dependen de los cálculos de distancia, como la agrupación y el análisis de componentes principales, las entidades con rangos de valores más grandes tienden a dominar las entidades con rangos más pequeños. Por lo tanto, tener características dentro de una escala similar nos permite comparar la importancia de las características y también ayuda a que los algoritmos converjan más rápido, mejorando así el rendimiento y los tiempos de entrenamiento.

1. Estandarización

También llamado escala de varianza, es el proceso de centrar los datos en cero y hacer las varianza uno. Para centrar una variable numérica, el promedio se resta de todos los valores. Como resultado del centrado, se tiene una media en cero. De manera similar, para escalar los datos, cada valor de la variable se divide por su desviación estándar. Escalar los datos coacciona los valores para que tengan una desviación estándar común de uno. Estas manipulaciones se utilizan generalmente para mejorar la estabilidad numérica de algunos cálculos.

Para estandarizar las observaciones, se resta la media a cada una de ellas y se divide por la desviación estándar

$$X' = \frac{X - \bar{x}}{\sigma}$$

donde $\bar{x}$ es la media aritmética y σ es la desviación estándar. El conjunto Z se dice estandarizado y tiene promedio cero y desviación estándar uno.

- Normaliza errores cuándo los parámetros de población son conocidos.
- Trabaja bien para poblaciones que están normalmente distribuidas.
- La única desventaja real de estas transformaciones es la pérdida de interpretabilidad de los valores individuales, ya que los datos ya no están en las unidades originales.

PYTHON

```
scipy.stats.zscore(X)
StandardScaler.fit_transform(X) # from sklearn.preprocessing
```

2. Normalización basada en la media

En esta normalización centramos la media de la variable en cero y cambiamos la escala de la distribución al rango de valores. Este procedimiento implica restar la media de cada observación y luego dividir el resultado por la diferencia entre los valores mínimo y máximo:

$$X' = \frac{X - \bar{x}}{\text{máx}(X) - \text{mín}(X)}.$$

Esta transformación da como resultado una distribución centrada en 0, con sus valores mínimo y máximo dentro del rango de -1 a 1.

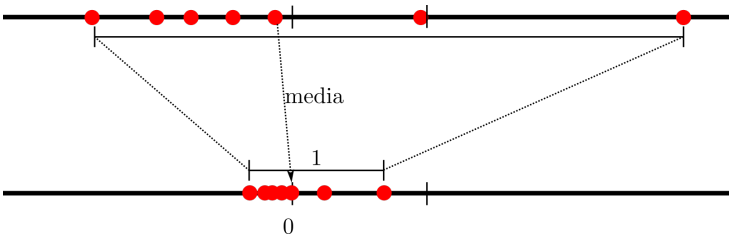

Figura 1: Normalización basada en la media.

PYTHON

```
RobustScaler.fit_transform(StandardScaler.fit_transform(X)) # from sklearn.preprocessing
  StandardScaler(with_mean=True, with_std=False)
  RobustScaler(with_centering=False,with_scaling=True,quantile_range=(0,100))
```

3. Escalamiento de valores máximo y mínimo

Sea x un valor de característica individual X, es decir, un valor de la característica en algún punto de los datos, y mín(x) y máx(x) los valores mínimo y máximo de esta característica en todo el conjunto de datos, respectivamente. El escalamiento mínimo-máximo comprime (o estira) todos los valores de características para que estén dentro del intervalo cerrado de $[0, 1]$.

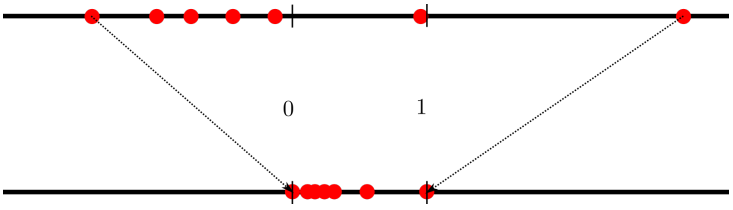

Figura 2: Escalamiento de valores máximo y mínimo.

Para implementar esta técnica de escalado, necesitamos restar el valor mínimo de todas las observaciones y dividir el resultado por el rango de valores, es decir, la diferencia entre los valores máximo y mínimo:

$$X' = \frac{X - \text{mín}(X)}{\text{máx}(X) - \text{mín}(X)}$$

PYTHON

```
MinMaxScaler.fit_transform(X) # from sklearn.preprocessing
```

4. Escalamiento de máximo absoluto

Esta transformación, mapea los valores de X hasta un máximo de 1. Esto se consigue dividiendo los datos entre el máximo:

$$X' = \frac{X}{\text{máx}(X)}$$

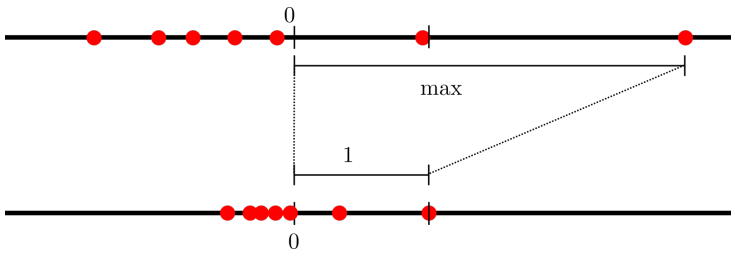

Figura 3: Escalamiento de máximo absoluto.

- Se recomienda usar esta transformación sobre datos centrados en cero o en un data–set con pocos datos.

PYTHON
MaxAbsScaler.fit_transform(X) # from sklearn.preprocessing

5. Escalamiento por cuantiles

Consiste en centrar las observaciones en cero usando la mediana y escalar el resultado por el rango intercuartílico (IQR)

$$X' = \frac{X - \bar{x}}{Q_3 - Q_1}$$

donde $\bar{x}$ es la mediana de X y Q_1, Q_3 corresponden a los cuartiles uno y tres.

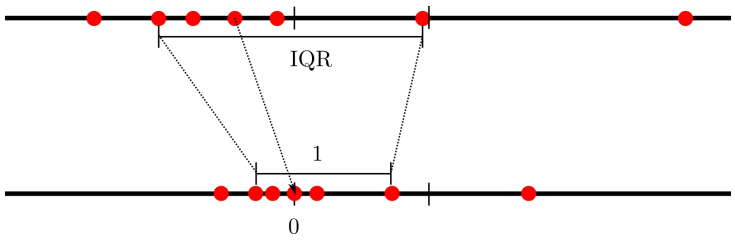

Figura 4: Escalamiento por cuantiles.

Este método se conoce como **escalamiento robusto** porque produce estimaciones más robustas para el centro y el rango de valores de la variable.

- Se recomienda si los datos contienen valores atípicos.

6. Criterio de elección

Dadas las características de cada normalización lo recomendable para comparar variables o los datos entre una variable es usar Estandarización, sin embargo, si la variable a estandarizar presenta valores atípicos, la normalización por cuantiles es robusta ante éstos. Ahora si la variable es usada como entrada para una función, es indispensable que el conjunto varíe en el rango de la función o que lo haga en la

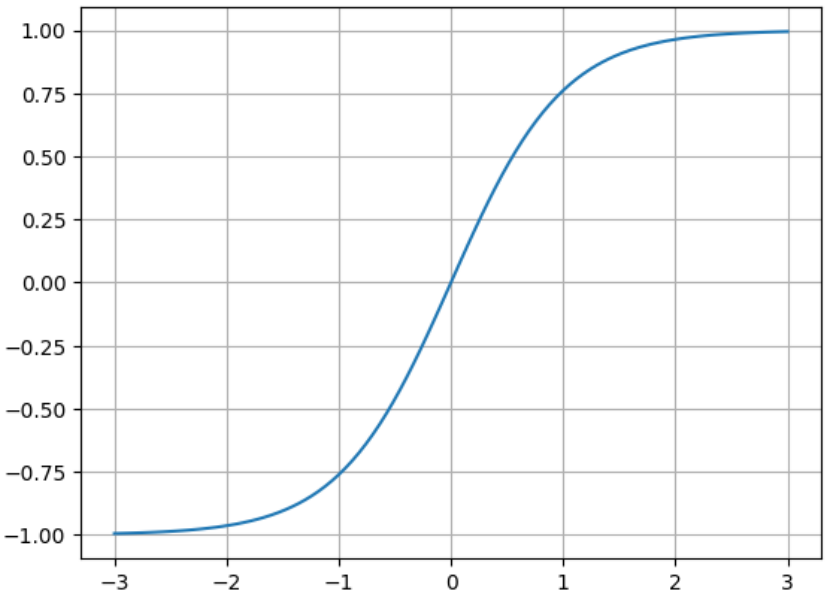

Figura 5: Función tangente hiperbólica.

Comparación entre variables	• Estandarización	La variable no presenta atípicos.
	• Por cuantiles	La variable presenta datos atípicos.
Evaluación en una función	• Máx–Mín	$x' \in [0, 1]$
	• Máx–Abs	$x' \in (-1, +1]$
	• con base en la media	$x' \in (-1, +1)$

Tabla 1: Criterio de elección para normalizaciones.

zona con mayor sensibilidad, por ejemplo, en una función como en la Figura 5, es deseable que el rango esté entre -1 y +1. En este caso, se puede usar el escalamiento de valores máximo y mínimo para obtener un rango exacto; escalamiento por máximo absoluto si la variable está centrada o se tienen pocos datos, en este último caso se asume una media cercana a cero (considere esta suposición); normalización basada en la media para obtener una variación simétrica. La Tabla 1 sintetiza lo anterior.

Referencias

[1] Zheng, A., & Casari, A. (2018). Feature engineering for machine learning: principles and techniques for data scientists. “O’Reilly Media, Inc.”.

[2] `https://es.wikipedia.org/wiki/NormalizaciÃ§n_(estadÃŋstica)`