UNIVERSIDAD EAFIT

Parcial 1

Ingeniería Matemática
Algebra en Ciencia de los Datos

Profesor:
Johan Felipe Garcia
Año Académico:
2023

Camilo Oberndorfer Mejía,
1000454952

1. Pregunta 1

Tomando la distancia de hamming entre dos vectores booleanos de lomgitud n es el numero de entradas que difieren

1. Demuestre que efectivamente es una distancia
2. Encuentre un conjunto de datos con al menos 5 variables booleanas, 4 de entrada y una objetivo. Entrene un algoritmo de clasificacion usando como metrica la distancia de Hamming.

1.1.

La distancia de Hamming entre dos vectores booleanos v y w de longitud n se define como:

$$d_H(v,w) = \sum_{i=1}^{n} |v_i - w_i|, \tag{1}$$

donde v_i y w_i son los elementos de los vectores v y w respectivamente, y $|x|$ representa el valor absoluto de x. Se puede notar que por la forma en la que esta definida esta distancia, es identica a la distancia l_1 o distancia de manhattan.

$$d_{l_1}(v,w) = \sum_{i=1}^{n} |v_i - w_i|, \tag{2}$$

Al ver que estas dos son definidas identicamente, entonces sabemos la distancia de hamming para booleanos es efectivamente una distancia.
Demostremos las tres condiciones para que una funcion pueda ser una distancia.

1. $d(x,y) \geq 0$, **y** $d(x,y) = 0 \Leftrightarrow x = y$: Dado la definicion de la distancia de hamming (1), este nunca podra ser negativo por ser la suma de valores absolutos. Y solo va a poder ser 0 si y solo si todos los componentes del vector v y el vector w son iguales, por tanto $v = w$.
2. $d(x,y) = d(y,x)$

$$d(x,y) = d((x_1,\dots x_n),(y_1,\dots,y_n)) = \sum_{i=1}^{n} |x_i - y_i| = \sum_{i=1}^{n} |y_i - x_i| = d((y_1,\dots y_n),(x_1,\dots,x_n)) = d(y,x)$$

3. $d(x,y) \leq d(x,z) + d(z,y)$ Para demostrar la desigualdad triangular de la distancia de Hamming, es suficiente mostrar que $d(a,b) \leq d(a,c)+d(c,b)$, ya que entonces también tenemos $d(a,c)-d(c,b) \leq d(a,b)$. En otras palabras, es suficiente demostrar que

$$d(a,b) = \sum_{i=1}^{n} |a_i - b_i| \leq \sum_{i=1}^{n} (|a_i - c_i| + |c_i - b_i|),$$

La suma de valores absolutos, siempre va a ser suma de valores positivos, por lo que $\sum_{i=1}^{n} |a_i - b_i| \leq \sum_{i=1}^{n} (|a_i - c_i| + |c_i - b_i|)$,. Ptor tanto la desigualdad se cumple en todos los casos.

1.2.

Se utilizo un dataset de atributos de diferentes animales.
Información de Atributos: (nombre del atributo y tipo de dominio de valores)

- **animal_name**: Único para cada instancia.
- **hair**: Booleano.
- **feathers**: Booleano.
- **eggs**: Booleano.
- **milk**: Booleano.
- **airborne**: Booleano.
- **aquatic**: Booleano.
- **predator**: Booleano.
- **toothed**: Booleano.
- **backbone**: Booleano.
- **breathes**: Booleano.
- **venomous**: Booleano.
- **fins**: Booleano.
- **legs**: Numérico (conjunto de valores: {0, 2, 4, 5, 6, 8}).
- **tail**: Booleano.
- **domestic**: Booleano.
- **catsize**: Booleano.
- **class_type**: Numérico (valores enteros en el rango [1, 7]).

Proceso de Entrenamiento de la Support Vector Machine (SVM)

El proceso de entrenamiento de la Support Vector Machine (SVM) se llevó a cabo en los siguientes pasos:

1. **Transformación de Datos**: Se realizó una transformación de la variable "legs.$^{\text{en}}$ una variable booleana, donde se asignó el valor 0 si el animal tenía 0 piernas y el valor 1 si tenía más de una pierna.
2. **Selección de Características**: Se utilizaron las demás variables booleanas, como "hair,feathers,eggs,.$^{\text{et}}$c., como características para el entrenamiento de la SVM.
3. **Entrenamiento de la SVM**: Se entrenó una SVM utilizando las características mencionadas. Como parte del proceso, se eligió la distancia de Hamming o manhattan (intercambiables en este caso) como kernel para la SVM. Esta elección de kernel se basó en la naturaleza de los datos y los objetivos del análisis.

Este proceso de entrenamiento permitió construir un modelo de SVM capaz de clasificar animales en diferentes categorías utilizando las características mencionadas y la distancia de Manhattan como métrica de similitud.
Al final se utilizo coom metrica de desempeño la distancia de hamming entre el vector predecido contra el vector real.
Se obtuvo una prediccion del 95.2 % de precision con esta metrica.