Instituto Tecnológico de Costa Rica

Campus Tecnológico Alajuela

Escuela de Ingeniería en Computación

Tarea 08

IC-6200 Inteligencia Artificial

Aprendizaje automático KNN

Profesora:
María Auxiliadora Mora Cross

Estudiantes:

Rodolfo Cruz Vega - 2013235955
Jonathan Quesada Salas - 2020023583

Semestre I 2023

Ejercicio 1: KNN

1) (3 puntos) Calcule la similitud entre todos los casos y un nuevo caso q con los siguientes valores de características (Alta, no, no, no).

Medida de similitud

Para fiebre (F)

	Casos en la base de casos		
Caso nuevo	No	Promedio	Alta
No	1.0	0.7	0.2
Promedio	0.5	1.0	0.8
Alta	0.0	0.3	1.0

Para vómito(V), Diarrea(D) y Escalofríos (E)

	Casos	
Caso nuevo	Sí	No
Sí	1.0	0.0
No	0.2	1.0

Valores de caso nuevo (q)

q	Alto	No	No	No

Base de casos

Caso	Fiebre (F)	Vómito (V)	Diarrea (D)	Escalofríos (E)	Diagnostico
c1	No	No	No	No	Saludable
c2	Promedio	No	No	No	Influenza

c3	Alta	No	No	Sí	Influenza
c4	Alta	Sí	Sí	No	Salmonela
c5	Promedio	No	Sí	No	Salmonela
c6	No	Sí	Sí	No	Inflamación intestinal
c7	Promedio	Sí	Sí	No	Inflamación intestinal

Cálculo de similitud entre q y c_i

Caso	Fiebre (F)	Vómito (V)	Diarrea (D)	Escalofríos (E)
c1	0	1	1	1
c2	0.3	1	1	1
c3	1	1	1	0.2
c4	1	0.2	0.2	1
c5	0.3	1	0.2	1
c6	0	0.2	0.2	1
c7	0.3	0.2	0.2	1

Pesos

Pesos	WF	WV	WD	WE
	0.3	0.2	0.2	0.3

Peso por atributo

Caso	Fiebre (F)	Vómito (V)	Diarrea (D)	Escalofríos (E)	Resta de Pesos ^2	Peso por atributo
c1	0	0.2	0.2	0.3	0.49	0.7
c2	0.09	0.2	0.2	0.3	0.3721	0.61
c3	0.3	0.2	0.2	0.06	0.0256	0.16
c4	0.3	0.04	0.04	0.3	0.0064	0.08
c5	0.027	0.2	0.04	0.3	0.263169	0.513
c6	0	0.04	0.04	0.3	0.1444	0.38
c7	0.09	0.04	0.04	0.3	0.0841	0.29

2) (1 puntos) Utilice el algoritmo de K Vecinos más cercanos para determinar el diagnóstico con k=1.

El caso c1 tendría el mayor peso ponderado de 0.7. Por lo tanto, utilizando el algoritmo de K Vecinos más cercanos con k=1 y considerando los pesos, el diagnóstico para el caso q (Alto, No, No, No) sería "Saludable", ya que el caso c1 es el más cercano y tiene el mayor peso ponderado.

3) (1 puntos) Utilice el algoritmo de K Vecinos más cercanos para determinar el diagnóstico con k=4

Los 4 casos más cercanos según la similitud ponderada son:

c1 con una similitud ponderada de 0.7
c2 con una similitud ponderada de 0.61
c6 con una similitud ponderada de 0.38
c5 con una similitud ponderada de 0.513

Examinando los diagnósticos de estos 4 casos más cercanos, encontramos que:

c1: Saludable
c2: Influenza
c6: Inflamación intestinal
c5: Salmonela

En este caso, 4 diagnósticos diferentes (Influenza, Saludable, Inflamación intestinal, Salmonella). No hay una mayoría clara en los votos de los casos más cercanos.

En esta situación, se puede utilizar un enfoque adicional para desempatar. Sin embargo, dado que no se especifica un método adicional, no se puede determinar con certeza el diagnóstico para el nuevo caso q utilizando el algoritmo de K Vecinos más cercanos con k=4 y los pesos dados.

4) (2 puntos) Investigue cómo se resuelven los empates a la hora de asignar una instancia a una clase particular.

Cuando se presentan empates en el algoritmo de K Vecinos más cercanos al asignar una instancia a una clase particular, existen diferentes enfoques comunes para resolverlos.

Uno de los enfoques más utilizados es utilizar la votación ponderada, donde se asignan pesos a los vecinos basados en su similitud o distancia. En este método, cada vecino contribuye a la decisión final con un peso proporcional a su similitud con la instancia a clasificar. Luego, se realiza una votación ponderada y la clase con mayor peso total se asigna como la clase predicha. Este enfoque permite tomar en consideración la relevancia de cada vecino en función de su similitud o distancia con la instancia en cuestión (Hastie et al., 2009).

Otro enfoque es utilizar la distancia promedio, donde en lugar de seleccionar el vecino más cercano, se calcula la distancia promedio entre la instancia a clasificar y los vecinos más cercanos de cada clase. La clase con la distancia promedio más baja se asigna como la clase predicha. Este método permite tener en cuenta la distribución de los vecinos cercanos en lugar de considerar solo un vecino individual (Mitchell, 1997).

En casos donde persiste un empate después de aplicar las técnicas anteriores, se pueden utilizar estrategias adicionales, como aumentar el valor de k (k ampliado) para incluir más vecinos en la votación y resolver el empate (Duda et al., 2012). También se puede recurrir a un orden lexicográfico, donde se establece una

prioridad de desempate basada en el orden de las clases, asignando la clase en orden alfabético en caso de empate (Zhang et al., 2019).

Ejercicio 2:

1) Lea la publicación: Saadatfar, H(2020). A New K-Nearest Neighbors Classifier for Big Data Based on Efficient Data Pruning. Recuperado de https://www.mdpi.com/2227- 7390/8/2/286/htm

2) (5 puntos) Comente en qué consiste el algoritmo LC-KNN que proponen los investigadores.

El algoritmo LC-KNN es una variante del algoritmo KNN que se utiliza para clasificar conjuntos de datos grandes. El algoritmo LC-KNN (Lazy Classifier - K-Nearest Neighbors) propuesto en el artículo "A New K-Nearest Neighbors Classifier for Big Data Based on Efficient Data Pruning" aborda el desafío de clasificar conjuntos de datos grandes de manera eficiente y precisa. Este algoritmo es una variante del conocido algoritmo K-Nearest Neighbors (KNN).

El KNN tradicional tiene una alta complejidad computacional, ya que requiere calcular la distancia entre la instancia de prueba y todas las instancias de entrenamiento. Para superar esta limitación, el algoritmo LC-KNN introduce una estrategia de poda de datos eficiente que reduce el número de instancias de entrenamiento consideradas durante la fase de clasificación.

La estrategia de poda se basa en el concepto de "clasificadores perezosos" (lazy classifiers), que aplazan el procesamiento de los datos de entrenamiento hasta el momento de la clasificación. En LC-KNN, se realiza una etapa de preprocesamiento para identificar y eliminar instancias redundantes y no informativas del conjunto de entrenamiento.

La estrategia de poda se basa en la detección de regiones densamente pobladas en el espacio de características. Estas regiones representan instancias que no aportan información adicional para la clasificación y, por lo tanto, pueden ser eliminadas. Al reducir el tamaño del conjunto de datos de entrenamiento, se logra una clasificación más rápida y precisa.

La eliminación de instancias redundantes también mejora la precisión de la clasificación al eliminar el ruido y los puntos atípicos. Esto es especialmente beneficioso en conjuntos de datos grandes, donde es más probable encontrar instancias redundantes y ruido.

El algoritmo LC-KNN utiliza una estrategia de poda de datos eficiente para mejorar la eficiencia y precisión de la clasificación de conjuntos de datos grandes. Al reducir el tamaño del conjunto de datos de entrenamiento y eliminar instancias redundantes, se logra una clasificación más rápida y precisa. Esto lo hace especialmente útil en el contexto de big data, donde la eficiencia computacional es un factor crítico.

Referencias:

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). Pattern Classification. John Wiley & Sons.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.

Mitchell, T. (1997). Machine Learning. McGraw Hill.

Zhang, T., Zhang, C., & Zhang, Z. (2019). A Comparative Study of k-Nearest Neighbor Algorithms for Classification. Information, 10(9), 280.