Práctica 2

Mikel de Velasco, Ieltzu Irazu, M Inés Fernandez

16 de octubre de 2015

1 Introducción

Se pide disear un clasificador que implemente el método k-NN básico con la distancia de Minkowski según la expresión (4) donde n representa el número de atributos empleados para caracterizar las muestras.

$$d(a,b) = \left[\sum_{i=1}^{n} |a_i - b_i|^m\right]^{\frac{1}{m}} \tag{1}$$

El clasificador permitirá seleccionar tanto el número de vecinos a explorar (k) como el parámetro m de la expresión (4). Elegir el lenguaje de programación que se considere más apropiado para el diseo. Para inferir el clasificador se dispone de un conjunto de datos (Diabetes.arff). El conjunto de datos dispone de 768 instancias para inferir el modelo. Para describir las instancias se utilizan 8 atributos más la clase. Los atributos son los siguientes:

- Number of times pregnant
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- Diastolic blood pressure
- Triceps skin fold thickness
- 2-Hour serum insulin
- Body mass index
- Diabetes pedigree function
- Age
- Class variable

La clase que hay que determinar es si el paciente tiene diabetes o no.

1.1 Parámetro k

El parámetro k es el parámetro que utilizaremos para indicar cuantos vecinos vamos a usar para evaluar el clasificador. Cuantos más usemos mejor, pero, también aumentara el costo del algoritmo.

1.2 Parámetro m

El parámetro m sera el parámetro que utilizaremos para saber que distancia tenemos que utilizar para evaluar nuestro clasificador.

En esta práctica hemos implementado 3 distintas distancias:

1. Distancia de **Manhattan**:

   La distanciade Manhattan (2), entre dos vectores **x**, **y** en un espacio vectorial real n-dimensional con un sistema de Coordenadas cartesianas fijo es la suma de las longitudes de las proyecciones del segmento de lnea entre los puntos sobre el sistema de ejes coordenados.

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i| \tag{2}$$

   Donde n es el número de dimensiones.

2. Distancia **Euclídea**:

   Un espacio euclídeo es un espacio vectorial formado sobre los nmeros reales de dimensin finita, en que la norma es la asociada al producto escalar ordinario. Para cada nmero entero no negativo n, el espacio euclídeo n-dimensional se representa por el smbolo $\mathbb{R}^n$ y es el conjunto de todas las tuplas ordenadas $(x_1, x_2, \ldots, x_n)$ en donde cada x_i es un número real, junto con la funcin distancia entre dos puntos $(x1, \ldots, x_n)e(y1, \ldots, y_n)$ definida por la frmula:

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3}$$

   Esta función distancia es una generalización del teorema de Pitágoras y se denomina Distancia euclidiana (3).

3. Distancia de **Minkowski**:

   En fsica matemtica, el espacio de Minkowski (o espacio-tiempo de Minkowski) es una variedad lorentziana de cuatro dimensiones y curvatura nula, usada para describir los fenmenos fsicos en el marco de la teora especial de la relatividad de Einstein.

   En el espacio de Minkowski pueden distinguirse tres dimensiones espaciales ordinarias y una dimensin temporal adicional, de tal manera que todas juntas forman una 4-variedad y as representar al espacio-tiempo.

El espacio-tiempo de Minkowski es una variedad lorentziana de curvatura nula e isomorfa a $\mathrm{M}_0 = (^4, \eta)dondeeltensormtricopuedell$
$-dx^0 \otimes dx^0 + dx^1 \otimes dx^1 + dx^2 \otimes dx^2 + dx^3 \otimes dx^3(4)$

2 Metodología

En este apartado desarrollaremos nuestro propio algoritmo K-Nearest Neighbors y describiremos el diseño y cómo hemos implementado este algoritmo. Además haremos una breve descripción de como hacer funcionar el programa.

El algoritmo que hemos desarrollado es básicamente una adaptación del clasificador IBk ya que hemos cogido su código y le hemos cambiado la forma de estimar que estaban utilizando.

Nuestro programa puede ser ejecutado desde eclipse de una manera muy fácil. Si ejecutas el Probador.java conseguirś una ejecución total del programa y obtendrás los resultados en la carpeta de ficheros. Por otro lado, también se pueden ejecutar de forma individual. Se diferenciarían 3 ejecuciones diferentes. Preprocesador.java nos sirve para quitar todo lo malo que venga en los datos(Extreme values, outliers,...).

Como parametros de entrada tendremos los siguientes

Después ejecutaríamos Modelo.java y con ello conseguiríamos obtener los dos modelos de los que estamos tratando en la practica.

Como parámetros de entrada tendremos los siguientes

Por ultimo, ejecutaríamos el Clasificador.java y con ello conseguiríamos los resultados que necesitamos para comparar los modelos anteriores.

Como parámetros de entrada tendremos los siguientes

3 Resultados

En esta sección mostraremos los resultados que hemos obtenido de las ejecuciones anteriores

4 Conclusiones

Realizar un clasificador desde cero ha sido una tarea complicada a que nos hemos basado en el modelo del IBk, y le hemos ajustado el pseudocódigo que teníamos. Ha sido una tarea entretenida y muy dinámica pero muy costosa. Como conclusión tenemos que decir que no merece la pena crear un modelo propio habiendo ya el IBk.

5 Valoración Subjetiva

Una vez finalizado todo el proyecto, cada uno hemos echo una valoración de todo lo que ha supuesto el proyecto individualmente.

Ieltzu: Personalmente ha sido una tarea que no me ha gustado mucho. He visto inecesaria la practica y para lo único que me ha servido es para darme cuenta de lo difícil que es implementar un clasificador propio.

Mikel: Bajo mi prespectíva, esta práctica, ha ayudado a comprender como funciona el algoritmo k-NN con los diferentes párametros pasados. Sabiendo que alcanzar la eficiencia del algoritmo ya implementado por weka es difícil (ya que este ha sido testeado y comprobado por numerosos investigadores) es un buen mtodo para aprender a utilizar herramientas que están bien testeadas ademas de comprender por dentro como funcionan dichas herramientas intentando implementarlas nosotros.

Maria:

References