

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Clasificación con K vecinos más cercanos (KNN)

Andrés G. Abad, Ph.D.

Agenda

Introducción a los modelos de k vecinos más cercanos (KNN)

Pre-procesamiento de datos

Algoritmo de KNN

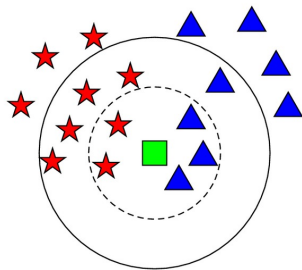
Introducción a los modelos de k vecinos más cercanos (KNN) I

- Dime con quien andas y te diré quien eres. . .



Introducción a los modelos de k vecinos más cercanos (KNN) II

- ▶ Un poderoso algoritmo no paramétrico de clasificación utilizado en reconocimiento de patrones
- ▶ KNN guarda todas las instancias disponibles y clasifica nuevas instancias basadas en una medida de similitud (e.g. una función de distancia)
- ▶ Se clasifica según la mayoría de votos entre las k instancias más cercanas



- ▶ Centrar y estandarizar
 1. Centrar: restar la media de cada vector
 2. Estandarizar: dividir para la desviación estandar
 - ⇒ $Mean = 0$ y $STDEV = 1$
 3. Centrar y estandarizar con la función `scale()`
- ▶ Transformación Log
- ▶ Transformación Ranking: se reemplazan los valores medidos por sus rankings
- ▶ No transformar

Ponderando las características I

- ▶ Pondere cada característica según su importancia para la clasificación

$$D(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_i w_k (x_i - y_i)^2}$$

- ▶ Podemos usar nuestro conocimiento a priori para decidir que características son más importantes
- ▶ Podemos aprender los pesos w_i utilizando *validación cruzada*

Normalización de características I

- ▶ Distancia puede ser dominada por unos atributos con valores relativamente grandes

$$x'_i = \frac{x_i - \min \{x_1, \dots, x_p\}}{\max \{x_1, \dots, x_p\} - \min \{x_1, \dots, x_p\}}$$

- ▶ Mapea los valores al rango $\mathbb{R}_{[0,1]}$
- ▶ Sucede cuando los features están en escalas muy diferentes

Noción de distancia entre objetos I

Toda noción de distancia entre dos objetos x y y debe cumplir los siguientes axiomas:

- ▶ $d(x, y) = d(y, x)$
- ▶ $d(x, x) = 0$
- ▶ $d(x, y) = 0$ si y solo si $x = y$
- ▶ $d(x, y) \leq d(x, z) + d(z, y)$

Noción de distancia entre objetos II

Las siguientes son distancias comunmente utilizadas en el análisis de datos:

- ▶ Variables numéricas
 - ▶ Distancia euclideana
 - ▶ Distancia manhattan
 - ▶ Distancia Canberra
 - ▶ Distancia Minkowski
- ▶ Variables binarias
 - ▶ Índice de Jaccard
- ▶ Variables categóricas
 - ▶ Índice de dice
- ▶ Distribuciones de probabilidad
 - ▶ Divergencia de Kullback-Leibler

Distancias para variables numéricas I

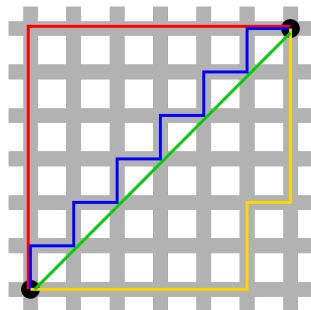
Considere dos vectores $\mathbf{x} = [x_1, \dots, x_p]$ y $\mathbf{y} = [y_1, \dots, y_p]$ pertenecientes a \mathbb{R}^p

- **Distancia euclídeana:**

$$l_2(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + \dots + (x_p - y_p)^2}$$

- **Distancia manhattan:**

$$l_1(\mathbf{x}, \mathbf{y}) = |x_1 - y_1| + \dots + |x_p - y_p|$$



Distancias para variables numéricas II

- **Distancia Canberra:**

$$d(\mathbf{x}, \mathbf{y}) = \frac{|x_1 - y_1|}{|x_1| + |y_1|} + \dots + \frac{|x_p - y_p|}{|x_p| + |y_p|}$$

- **Distancia Minkowski:**

$$l_p(\mathbf{x}, \mathbf{y}) = \left(|x_1 - y_1|^p + \dots + |x_p - y_p|^p \right)^{1/p}$$

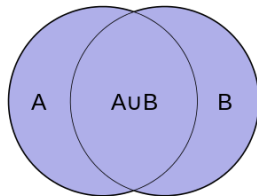
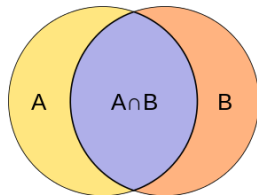
Distancia para variables binarias I

- ▶ **Índice Jaccard:** útil para vectores binarios de presencia o ausencia
- ▶ Mide la similitud entre dos muestras finitas

$$J(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} = \frac{|X \cap Y|}{|X| + |Y| - |X \cap Y|}$$

- ▶ Considere dos vectores
 $\mathbf{x} = [x_1, \dots, x_p]$ y $\mathbf{y} = [y_1, \dots, y_p]$
donde $x_i, y_j \in \{0, 1\}$

$$J(\mathbf{x}, \mathbf{y}) = \frac{\sum_i x_i y_i}{\sum_i x_i + \sum_j y_j - \sum_k x_k y_k}$$



Distancia para variables binarias II

Distancia basada en la correlación: $1 - r$

- Coeficiente de correlación de Pearson (PCC)

$$r = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{\sqrt{(\sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2)(\sum_{i=1}^n y_i^2 - (\sum_{i=1}^n y_i)^2)}}$$


Desventaja: sensitivo a *outliers*

Distancia para variables categóricas I

- **Índice Dice:** útil para variables categóricas primero convertidos en binarias)

$$QS(X, Y) = \frac{2|X \cap Y|}{|X| + |Y|}$$

Color	Red	Red	Yellow	Green	Yellow
-------	-----	-----	--------	-------	--------



Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1

Distancia para distribuciones de probabilidad I

- **Divergencia Kullback-Leibler:** útil para comparar distribuciones de probabilidad

$$D_{KL}(P||Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

- No es una distancia debido a que no es simétrica y no respeta la desigualdad triangular

Combinando distancias - Distancia de Gower I

La **distancia de Gower** es un método para combinar en una sola medida de distancia varias variables de tipos diferentes (numéricas, binarias, categóricas, etc)

Distancia de Gower

1. Asigne una medida de distancia para cada variable y luego normalícela entre 0 y 1
2. Combine (convexamente) las diferentes medidas; generalmente utilice el promedio

Combinando distancias - Distancia de Gower II

En detalle, la distancia de Gower d_G entre dos objetos $\mathbf{x} = [x_1, \dots, x_p]$ y $\mathbf{y} = [y_1, \dots, y_p]$ es la siguiente.

1. Considere las distancias (parciales) d_1, \dots, d_p , donde $d_i = d(x_i, y_i)$.
2. Normalice las distancias haciendo $\tilde{d}_i = d_i / (\max_{\mathbf{x}, \mathbf{y}}(d_i) - \min_{\mathbf{x}, \mathbf{y}}(d_i))$
3. Combine estas distancias parciales normalizadas haciendo $d_G = \sum_i \lambda_i \tilde{d}_i$, donde $\sum_i \lambda_i = 1$

Algoritmo de KNN I

- ▶ Es un método del aprendizaje basado en instancias
- ▶ Todas las instancias corresponden a puntos en un espacio n -dimensional
- ▶ Cada punto en el *training set* consiste en un conjunto de vectores y una etiqueta con la clase asociada
- ▶ Se recomiendan menos de 20 características

Vecino más cercano

- ▶ Dada una instancia \mathbf{x}_q , primero encuentre en el *training set* la instancia más cercana \mathbf{x}_l y estime

$$\hat{f}(\mathbf{x}_q) \leftarrow f(\mathbf{x}_l)$$

Algoritmo de KNN II

K vecinos más cercanos (KNN)

- ▶ Dada una instancia \mathbf{x}_q , vote entre los K vecinos más cercanos (si respuesta es discreta)
- ▶ Toma el promedio del valor f de los K vecinos más cercanos

$$\hat{f}(\mathbf{x}_q) \leftarrow \frac{\sum_{k=1}^K f(\mathbf{x}_k)}{K}$$

Algoritmo de KNN III



Cliente	Edad	Ingresos	# tarjetas de cred.	Acepta
Jorge	35	35k	3	No
Raquel	22	50k	2	Si
Ricardo	63	200k	1	No
Tomás	59	170k	1	No
Ana	25	40k	4	Si
Juán	37	50k	2	?

Seleccionando el valor de K

- ▶ El valor de K tiene un fuerte efecto en el desempeño de KNN
 - ▶ valor grande: todo se clasifica como la probabilidad *a priori* mayor
 - ▶ valor pequeño: alta variabilidad, borde de clasificación inestables
 - ▶ Pequeños cambios en el *training set* grandes cambios en los resultados de clasificación
 - ▶ Afecta la “suavidad” del borde de clasificación
- ▶ Seleccionando el valor de K
 - ▶ cree el *validation set* convendiendo una porción del *training set*
 - ▶ varíe K considerando el error de validación
 - ▶ Alternativamente, utilice la *rule of thumb* de $K < \sqrt{n}$, donde n es el número de instancias

KNN ponderado I

Podríamos desear ponderar más a los vecinos más cercanos

$$\hat{f}(\mathbf{x}_q) \leftarrow \frac{\sum_{k=1}^K w_k f(\mathbf{x}_k)}{K},$$

donde

$$w_k = \frac{1}{d(\mathbf{x}_q, \mathbf{x}_k)^2},$$

y $d(\mathbf{x}_q, \mathbf{x}_k)$ es la distancia entre \mathbf{x}_q y \mathbf{x}_k .

Note como ahora puede tener sentido utilizar *todas* las instancias en el conjunto de entrenamiento.

Fortalezas y debilidades I

Fortalezas

- ▶ Muy simple e intuitivo
- ▶ Entrenamiento rápido
- ▶ Puede ser aplicado a datos de cualquier distribución
- ▶ Buen clasificador si el número de instancias es lo suficientemente grande

Debilidades

- ▶ Lento al momento de clasificar nuevas instancias
 - ▶ Necesita calcular y comparar distancias de la nueva instancia a todas las demás
- ▶ Escoger un buen valor de K puede ser difícil
- ▶ Necesita un número grande de instancias para buena precisión