

Memoria P1 Anton Oellerer

Anton Oellerer

2017-10-06

1 Simultad coseno recursivo

1.1 Pseudocódigo

- **Entrada:** vectores x, y
- **Salida:** la simultud coseno de de los vectores, calculado recursivo
- **Procesamiento:** Si los vectores estan validos (no nil y no solo 0s) calcula el productode los vectores recursivo. Despues dividie el producto por el producto de los raizes de los cuadrados de los vectores.

1.2 Ejemplos

```
(sc-rec nil nil) -> nil
(sc-rec (0 0) (1 2)) -> nil
(sc-rec (1 1) (1 1)) -> 1
(sc-rec (-1 -1) (1 1)) -> -1
(sc-rec (1 0) (0 1)) -> 0
(sc-rec (5 7) (4 3)) -> 0.95
```

1.3 Comentarios

- A causa de errores de redondear, puede ser que los resultados no son exactos (por ejemplo 1.0000001 en lugar de 1)
- La recursion multiplica los primeros dos elementos de los vectores y recurso en el resto hasta no hay mas y en el camino de vuelta summa los productos
- Cuando un vector est nil o solo de 0s, nil esta retonado
- Los cuadrados son calculado por el producto de el vector con su mismo

2 Simultad coseno con mapcar

2.1 Pseudocódigo

- **Entrada:** vectores x, y
- **Salida:** la simultad coseno de de los vectores, calculado con mapcar
- **Procesamiento:** Si los vectores estan validos (no nil y no solo 0s) calcula el productode los vectores con mapcar. Despues dividie el producto por el producto de los raizes de los cuadrados de los vectores.

2.2 Ejemplos

Como recursivo

2.3 Comentarios

- A causa de errores de redondear, puede ser que los resultados no son exactos (por ejemplo 1.0000001 en lugar de 1)
- El producto de los vectores es calculado por una reduccion con '+' sobre un mapcar que multiplica todas las parejas de los vectores
- Cuando un vector est nil o solo de 0s, nil esta retonado
- Los cuadrados son calculado por el producto de el vector con su mismo

3 sc-conf

3.1 Pseudocódigo

- **Entrada:** vector cat, vector de vectores vs, float conf
- **Salida:** Vectores cuya simultad a cat es superior a conf, ordenados
- **Procesamiento:** Por cada vector de vs, calcula la simultad coseno con cat, despues excluye los resultados que son mas pequeños que la confianza y ordena la lista

3.2 Ejemplos

(sc-conf '(1 2 3) '((1 2 3) (0 1 0) (4 5 6)) 0.7) -> (0.97463 0.99999)

3.3 Comentarios

- Como con los otros, errores de redondear eran posible

4 sc classifier

4.1 Pseudocódigo

- **Entrada:** vectores de vectores cats, texts; función func
- **Salida:** Por cada texto, la categoría que tiene el similitud coseno máximo y ese resultado)
- **Procesamiento:** Por cada vector de texto, calcula todas las scs con la función indicada por las categorías y retorna el identificador y el resultado de la categoría con sc máxima.

4.2 Ejemplos

```
(sc-classifier '((0 1 2 3 4) (1 3 2 2 0) (2 3 5 8 0)) '((0 1 2 3 4) (1 3 2 1 0)))
```

4.3 Comentarios

Como puede observar, calcular las scs es más rápido con mapcar, pero necesita más espacio.