Assignment 9

CS 532: Introduction to Web Science

Spring 2018

Chandrasekhar Reddy Muthyala

1

Question

```
1.  Using the data from A8:

- Consider each row in the blog-term matrix as a
1000 dimension vector, corresponding to a blog.

- From chapter 8, replace numpredict.euclidean() with
cosine as the distance metric. In other words, you'll
be computing the cosine between vectors of 1000 dimensions.

- Use knnestimate() to compute the nearest neighbors for both:

http://f-measure.blogspot.com/
http://ws-dl.blogspot.com/

for k={1,2,5,10,20}.
```

Answer

To solve this question I used the blog data I previously received in assignment 7 **blogdata.txt** and the code provided by the Programming Collective Intelligence book [2].

Euclidean distance is probably the one that you are most familiar with; it is essentially the magnitude of the vector obtained by subtracting the training data point from the point to be classified. Another common metric is Cosine similarity. Rather than calculating a magnitude, Cosine similarity instead uses the difference in direction between two vectors [4].

$$similarity = COS(\Theta) = \frac{A.B}{||A||||B||}$$

To calculate the cosine distance metric **numpredict.py** had to be modified to accompany this new function. The methods that changed in this file were the *knnestimate* and *getdistances* functions. In the *getdistances* function I simply swapped out the euclidean function for cosine as shown in Listing 1. In the *knnestimate* function and return a list of sorted distances in descending order.

I then went filtered the F-measure and Web Science research group's blogs and create separate vectors with their values. These vectors were used for the cosine measurement against the vector with all blogs.

```
1
2  def cosine(v1, v2):
3      '''
4      cosine function for 1000 blog matrix. zip v1, v2
5      for array of tuples
6      '''
7      sumab = sum([a * b for a, b in zip(v1, v2)])
8      suma = sum([a * a for a in v1])
9      sumb = sum([b * b for b in v2])
10     return sumab / math.sqrt(suma * sumb)
11
12
13 def getdistances(data, vec1):
14     distancelist = []
15
16     # Loop over every item in the dataset
17     for i, vals in enumerate(data):
18         vec2 = vals
19
```

```
20          # Add the distance and the index
21          distancelist.append((cosine(vec1, vec2), i))
22
23      # Sort by distance
24      distancelist.sort(reverse=True)
25      return distancelist
26
27
28  def knnestimate(data, vec1, k=5):
29      # Get sorted distances
30      distances = getdistances(data, vec1)
31      return distances
32
33
34  def inverseweight(dist, num=1.0, const=0.1):
35      return num / (dist + const)
36
37
38  def subtractweight(dist, const=1.0):
39      if dist > const:
40          return 0
```

Listing 1: Python script with included cosine function and knnestimate changes

```
1
2  from numpredict import *
3
4
5  def estimate(vectorValues, vectorBlog1, vectorBlog2):
6      nn = knnestimate(allBlogVector.values(), vectorBlog1)
7      print("\n\n************\tF-Measure\t************")
8      kvals = [1, 2, 5, 10, 20]
9      for k in kvals:
10         print('k =', k)
11         for j in range(k):
12             print('%s' % (list(allBlogVector.keys())[nn[j][1]]))
13         print("--------" )
14
15     print("************\tWeb Science and Digital Libraries
           Research Group\t************")
16     nn = knnestimate(allBlogVector.values(), vectorBlog2)
17     for k in kvals:
18         print('k =', k)
19         for j in range(k):
20             print('%s' % (list(allBlogVector.keys())[nn[j][1]]))
21         print("--------" )
22
23
```

```
24
25 def formatData():
26     givenBlog_1 = 'F-Measure'
27     givenBlog_2 = 'Web Science and Digital Libraries Research
           Group'
28     allBlogVector = {}
29     blog1List = []
30     blog2List = []
31     with open("blogdata.txt", 'r') as f:
32         allLines = f.readlines()
33         for i, line in enumerate(allLines):
34             if i == 0:
35                 # skip header
36                 continue
37             blogMatrix = line.strip().split('\t')
38             if blogMatrix[0] == givenBlog_1:
39                 for i in range(1, len(blogMatrix)):
40                     blog1List.append(float(blogMatrix[i]))
41             elif blogMatrix[0] == givenBlog_2:
42                 for i in range(1, len(blogMatrix)):
43                     blog2List.append(float(blogMatrix[i]))
44             else:
45                 allBlogVector[blogMatrix[0]] = []
46                 for i in range(1, len(blogMatrix)):
47                     allBlogVector[blogMatrix[0]].append(float(
                           blogMatrix[i]))
48     return allBlogVector, blog1List, blog2List
49
50
51 if __name__ == "__main__":
52     allBlogVector, vectorBlog1, vectorBlog2 = formatData()
53     estimate(allBlogVector.values(), vectorBlog1, vectorBlog2)
```

Listing 2: Python script to find KNN values

OutPut:

```
1
2
3  ************     F-Measure        ************
4  ('k =', 1)
5  DaveCromwell Writes
6  ---------
7  ('k =', 2)
8  DaveCromwell Writes
9  The Jeopardy of Contentment
10 ---------
```

```
('k =', 5)
DaveCromwell Writes
The Jeopardy of Contentment
Pithy Title Here
KiDCHAIR
Alex Denney
--------
('k =', 10)
DaveCromwell Writes
The Jeopardy of Contentment
Pithy Title Here
KiDCHAIR
Alex Denney
Skiptrack music
New Amusements
My Name Is Blue Canary
The Slow Music Movement
.
--------
('k =', 20)
DaveCromwell Writes
The Jeopardy of Contentment
Pithy Title Here
KiDCHAIR
Alex Denney
Skiptrack music
New Amusements
My Name Is Blue Canary
The Slow Music Movement
.
Diagnosis: No Radio
The Power of Independent Trucking
Did Not Chart
Captain Panda's Local & Independent Music Showcase
PSI LAB
unter diesem gesichtspunkt
Lyrically Speaking
macthemost
Notes from a Genius
She May Be Naked
--------
**********    Web Science and Digital Libraries Research Group
            **********
('k =', 1)
unter diesem gesichtspunkt
--------
('k =', 2)
unter diesem gesichtspunkt
Myopiamuse
```

```
--------
('k =', 5)
unter diesem gesichtspunkt
Myopiamuse
She May Be Naked
Alex Denney
DaveCromwell Writes
--------
('k =', 10)
unter diesem gesichtspunkt
Myopiamuse
She May Be Naked
Alex Denney
DaveCromwell Writes
Pithy Title Here
The Power of Independent Trucking
The Professional Daydreamer
Diagnosis: No Radio
My Name Is Blue Canary
--------
('k =', 20)
unter diesem gesichtspunkt
Myopiamuse
She May Be Naked
Alex Denney
DaveCromwell Writes
Pithy Title Here
The Power of Independent Trucking
The Professional Daydreamer
Diagnosis: No Radio
My Name Is Blue Canary
Mile In Mine
Morgan's Blog
Media Coursework
.
macthemost
New Amusements
ELLIA TOWNSEND A2
hello my name is justin.
PSI LAB
Nothing But Ordinary Glances At Extraordinary Things
--------
```

Listing 3: Output of the knnestimate for two given blogs

References

[1] https://github.com/cmuth001/anwala.github.io/tree/master/Assignments/A7.

[2] Segaran, Toby. "Programming Collective Intelligence". O' Reilly, 2007. Web. 6 April 2017. http://shop.oreilly.com/product/9780596529321.do.

[3] https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[4] (https://towardsdatascience.com/introduction-to-k-nearest-neighbors-3b534bb11d26)