E-402-STFO PROBLEMS FOR MODULE E

CREATED BY HENNING ULFARSSON

This module is concerned with k-nearest neighbors.

You get a perfect score for this module by getting 65 points or more.

Problem 1 (1 point)**.** Write a function `mEp1(x,y)` that given two vectors `x` and `y` of the same length, computes the Euclidean distance between `x` and `y`.

```
Input: x = vector([1,2,3])
       y = vector([0,-1,0.5])
Run: mEp1(x,y)
Output: 4.03
```

Problem 2 (1 point)**.** Write a function `mEp2(x,y)` that given two vectors `x` and `y` of the same length, computes the Manhattan distance between `x` and `y`.

```
Input: x = vector([1,2,3])
       y = vector([0,-1,0.5])
Run: mEp2(x,y)
Output: 6.5
```

Problem 3 (2 points)**.** Write a function `mEp3(x,y)` that given two vectors `x` and `y` of the same length, containing only `0`'s and `1`'s, computes the Hamming distance between `x` and `y`.

```
Input: x = vector([1,0,1,0])
       y = vector([1,1,0,1])
Run: mEp3(x,y)
Output: 3
```

Problem 4 (7 points)**.** Write a function `mEp4(x,y)` that given two vectors `x` and `y` of the same length, containing only `0`'s and `1`'s, computes the Levenshtein distance between `x` and `y`.

```
Input: x = vector([1,0,1,0])
       y = vector([1,1,0,1])
Run: mEp4(x,y)
Output: 2
```

Problem 5 (4 points)**.** Write a function `mEp5(x,y)` that given two vectors `x` and `y` of the same length, which is a square, computes the rank distance between the matrices constructed from the vectors.

```
Input: x = vector([1,0,1,1,1,1,0,0,1])
       y = vector([1,17,1,0,0,1,0,1,1])
Run: mEp5(x,y)
Output: 2
```

Date: Updated November 7, 2013.

Problem 6 (15 points)**.** Write a function `mEp6(L,dist)` that given a list `L` of tuples, `(P,a)`, where `P` is a point (or vector) with `n` coordinates, and `a` is an integer representing the label of the point `P`, checks whether the points in `L` satisfy the axiom of neighborliness with respect to the distance `dist`.

```
Input: L = [ ( (1,1,1),  2 ),\
             ( (0,0,0),  1 ),\
             ( (1,0,1),  2 ),\
             ( (0,0,-1), 1 ) ]
       dist = mEp1
Run: mEp6(L,dist)
Output: True
```

Problem 7 (15 points)**.** Write a function `mEp7(L,J,dist)` that given a list `L` of tuples, `(P,a)`, where `P` is a point (or vector) with `n` coordinates, and `a` is an integer representing the label of the point `P`, and another list `J`, containing points `Q` that need to be labelled, outputs a labelled version of `J`, using the nearest (in the metric `dist`) labelled neighbor of each `Q` to guess the label. Ties should be broken by choosing the lower valued label.

```
Input: L = [ ( (1,1,1),  2 ),\
             ( (0,0,0),  1 ),\
             ( (1,0,1),  2 ),\
             ( (0,0,-1), 1 ) ]
       J = [ (0,0,1) ]
       dist = mEp1
Run: mEp7(L,J,dist)
Output: [ ( (0,0,1), 1 ) ]
```

Problem 8 (10 points)**.** Write a function `mEp8(L,J,dist,k)` that given a list `L` of tuples, `(P,a)`, where `P` is a point (or vector) with `n` coordinates, and `a` is an integer representing the label of the point `P`, and another list `J`, containing points `Q` that need to be labelled, outputs a labelled version of `J`, using the `k` nearest (in the metric `dist`) labelled neighbor of each `Q` to guess the label (so the most common label gets chosen. Ties should be broken by choosing the lower valued label).

```
Input: L = [ ( (1,1,1),  2 ),\
             ( (0,0,0),  1 ),\
             ( (1,0,1),  2 ),\
             ( (0,0,-1), 1 ) ]
       J = [ (0,0,1) ]
       dist = mEp1
       k = 3
Run: mEp8(L,J,dist,k)
Output: 2
```

Problem 9 (20 points)**.** Repeat the exploration done on the page `http://jeremykun.com/2012/08/26/k-nearest-neighbors-and-handwritten-digit-classification/` on the same data (always use the first half to train on, **do not** shuffle the data) for all the metrics above and find which k is best for each. Write a short LaTeX-ed report.

School of Computer Science, Reykjavik University, Menntavegi 1, 101 Reykjavík, Iceland

E-mail address: `henningu@ru.is`