Program for Task2

Presenter: GUO Feng

guofeng314@163.com
Machine Learning Center
Hebei University, Baoding City,
Hebei Province, P.R. China
What is the Problem?

The problem is about indoor location estimation from radio signal strengths received by a client device from various WiFi Access Points (APs)

• At one location of 247, the client device can receive radio signal from some APs of 100
• There are 2322 collections at time A, but 1701 collections of them are given without location labels
• There are 53 collections at time B. all these collections are given with location labels, using as benchmarks
• There are 3128 collections at time B, needing predicting the location
The Challenges

- 247 classes
  - multi-class learning problem
- Only 621 instances of 2322 have class labels
  - semi-supervised learning
- Training and testing data are obtained at different time
Our Solution for Task2

- $k$-means method
- Assign the different importance of one attribute for different classes
- Nearest Unconsistent Neighbor distance
Step 1

Convert data type from text format to record format.

**text format**
-1  2:-93  10:-52  16:-86  18:-93  19:-95
124  18:-90  23:-73  45:-92  48:-90  59:-89  66:-88  67:-95

**record format**
-100 -100 -93 -100 -100 … -1
-100 -100 -100 -100 -100 … 124

"Task_2_training_data.txt" → "training_data",
"Task_2_landmark_data.txt" → "landmark_data"

Split the data in the training_data into two parts:
"location-known-data" and "location-unknown-data"
Step 2

- Compute the average of condition attribute values in "training_data" (≠ -100)
- Compute the average of condition attribute value of "landmark_data" (≠ -100)
- Compute the difference of the two averages denoted by "difference"
Step 3

• Subtract "difference" from the condition attribute value of "landmark_data" (≠ -100)

• The result is denoted by "LA_data"

• " + " landmark_data

• " x" LA_data
Step 4

• Make n copies of the "LA_data"
• Merge the n copies and "location-known data" into the training data set, denoted by "train-data"
• Merge "LA_data" and "location-known data" into the "old-data"
LA_data \rightarrow \ldots \rightarrow \text{location-known data} \rightarrow \text{train-data} 

LA_data \rightarrow \text{location-known data} \rightarrow \text{old-data}
Why?

• In my opinion, the instances of Landmark_data are more possible near the centers.

• To a certain extent, the more instances of this kind, the better estimation of the centers.
Step 5
Compute 247 centers of corresponding classes using "train-data"
Step 6

- Assign decision attribute of every record in "location-unknown data", which is equal to the nearest center by Euclidean distance.
- The result of the labeled "location-unknown data" is denoted by "location-unknown-labeled-data"
Label Location
Step 7

- Merge the "train-data" and the "location-unknown-labeled-data" into the "train-data"
Step 8

• Run Step 5:

Center changed!

Center changed!
Step 9

• Compute the NUN-distances of every record of the "old-data"

• NUN-distance is the shortest distance between the record and other records in different classes
NUN-distance
Step 10

- Convert the data in "Task_2_test_data.txt" from text format to record format, decision attributes are assigned -1
- The result is denoted by "datar"
Step 11

• Get the index of condition attributes of every center obtained in Step 8
• The index will be used in Step 13
Step 12

- Get the class centers of time B, which are equal to the centers of time A adding the "difference"
Step 13

- Calculate the Minkowski distance between every record of the "datar" and every center
- The Minkowski distance's weights of different centers is relevant to the condition index of every center
- Make decision attribute of every record of "datar" equal to the nearest center temporarily, which may be modified by step 14
Example of Computing Distance

• The i-th center is \((c_{i1}, c_{i2}, c_{i3}, c_{i4}, \ldots, c_{i100}, i)\),
  weights: \(w_{i1}, w_{i2}, w_{i3}, w_{i4}, \ldots, w_{i100}\);

• The j-th center is \((c_{j1}, c_{j2}, c_{j3}, c_{j4}, \ldots, c_{j100}, j)\),
  weights: \(w_{j1}, w_{j2}, w_{j3}, w_{j4}, \ldots, w_{j100}\);

• \(A(a_1, a_2, a_3, a_4, \ldots, a_{100}, -1)\) is a record of “datar”

• The distance between A and the i-th center

\[
= \sqrt{\sum_{k=1}^{100} (w_{ik} \times (C_{ik} - a_k)^2)}
\]

• The distance between A and the j-th center

\[
= \sqrt{\sum_{k=1}^{100} (w_{jk} \times (C_{jk} - a_k)^2)}
\]
Why is This Kind of Distance?

• Because the client device more possibly receives signals from the same APs at the same location

• If the signals are received at same location, the order of RSS values are more possible same

• The distance between two instance must be long, if they are different
If a green point is between two classes, the region of class red is bigger than the blue ($d_1 > d_2$), so the green point is assigned class blue. But the green point is nearer to one of the red points, the decision attribute of the green point should be equal to the red. Therefore, it needs modifying one record in "datar".
Step 14

- Calculated the Euclidean distance between every record in "datar" and every record of "old-data" which has added the "difference"
- For every record of the "datar", the nearest neighbor in the "old-data" is obtained
- If the Euclidean distance is smaller than $1/m$ of the neighbor's NUN-distance, make the decision attribute equal to the neighbor's decision attribute
Step 15

Output the decision attributes of every record of "datar" to text file "guofeng314@163.com_task2.txt"
Acknowledgments

• Thank my supervisor Prof. Xizhao Wang, who give many advices!
• Thank every one of Machine Learning Center who help me!
• Thank Dr. Gang Kou and Dr. Qiang Yang for swift replies!
• Thank Peng Zhang and Juan Qi!
Thank you!