MACHINE LEARNING – FALL 2014
PROJECT – SENSOR NETWORK DATA
MAPPING TEMPERATURE ZONES

VIVEK ANAND SAMPATH

1. INTRODUCTION:

The sensor network data has temperatures measured at 54 sensor nodes in the facility, the aim of this project is to map the regions by the predominant temperature recorded at each sensor location. By identifying this we will be able to propose air-conditioning solutions for the facility. The data set which contains temperature reading for every 30 seconds is passed to k-means clustering algorithm and the clusters are identified, the sensor locations are then assigned to appropriate clusters and the regions are plotted as a scatter graph using a color map coded from red to blue, red to mark the hotter regions and blue for colder regions.

2. PROBLEM DEFINITION AND ALGORITHM:

i. Task Definition:

The sensor network data is collected from 54 sensors and is a “real” dataset with lots of missing data, noise, and failed sensors giving out-lier values, especially when battery levels are low.

The dataset was chosen from below web page:
http://www.cs.cmu.edu/~guestrin/Class/10701/projects.html

Dataset was downloaded from *http://www.cs.cmu.edu/~guestrin/Research/Data/*.

The reading taken by sensors are presented in the dataset as:
time nodeid temperature humidity light voltage

The position of each sensor based on nodeid is provided in separate file in the below format:
nodeid x_location y_location

If we can identify the clusters for each of the parameter recorded by the set of all sensors, then it would be efficient to store the cluster data instead of storing all the raw data recorded by each of the sensors.

Though this algorithm is run on a sensor data recorded withing facility, this approach can easily be used for temperature recordings for vast regions like city, country.

ii. Algorithm Definition:

The class of problem that is studied is an unsupervised learning algorithm, we are provided with sensor data for a window of time and we are required to find the pattern in the temperature recorded at the facility. Clustering algorithms are used for these purpose, we will be using K-means clustering algorithm.

K-Means clustering algorithm:

Initialization:

From the temperature readings, chose k values and initialize the cluster values as those k values.

K-Means algorithm consists of two steps: 1) update assignments 2) recalculate centroid points.

1) Update assignments:

For each temperature reading, calculate the Euclidean distance w.r.t each cluster. Assign the data point to which is closest to.

2) Recalculate centroid points:

Now for all the data points belonging to same cluster, find mean value of the temperature readings and make it as cluster value of that cluster. Now we would have recalculated cluster points for all the clusters

Repeat these two above steps till all the data points didn't change clusters between two iterations. Now, the clusters would have converged and temperature readings are split into K different clusters.

Based on cluster assigned for each reading taken at a particular sensor node, we assign a cluster to the sensor node. The sensor node is assigned the cluster which was assigned the most times, for the readings taken by that sensor. For eg. If the sensor nodeid 10 has taken 10 readings, 3 of which was classified as cluster i, 2 of them as cluster j and 5 of them as cluster k, then sensor nodeid 10 is assigned to cluster k.

iii. Implementation:

Language used: Python

Plotting library: numpy, matplotlib

iv. Displaying the cluster regions:

The sensor location co-ordinates are provided as part of data set. The sensor nodeids are marked on a graph using the co-ordinates given and a scatter plot is made for the clusters associated with the sensor node.

The color map varies from blue to red. “Blue” signifies the region coldest and “Red” signifies the hottest region.

The area of each sensor node plot signifies the percentage of recordings supporting the current cluster assigned to the total readings recorded from that sensor node.

3. EXPERIMENTAL EVALUATION:

i. Varying number of clusters K:

The number of clusters has been varied for different values of k of 2, 5, 10, 15, 20, 25 and the results of each of those cases as provided as evidence.

ii. Varying the initialization points:
K-Means algorithm convergence greatly depends on the initialization points, so the experiment has been repeated several times and the result which conforms to maximum number of appearances are considered.

iii. Varying the sensor node assigning method:
Several strategies for assigning cluster to a sensor node based on the cluster assigned to the different readings taken by that sensor node was considered.

Some of the evaluation methods considered were:
1) Assigning the closest cluster to the mean of the readings recorded by that node
2) Assigning the cluster which was assigned to the readings the maximum number of times
3) Displaying the proportion of association with each cluster.

4. RELATED WORK:
Distributed Regression: an Efficient Framework for Modeling Sensor Network Data[1]
The sensor network data was analyzed so as to minimize the communication messages exchanged between the sensor nodes but at the same time not compromising on the data gathered, the trick there is to learn the parameters for the data recorded at each node and exchanging only the parameters learned rather than communicating the raw data. Further communication optimization was analyzed by sending readings from a single sensor node if group of neighboring sensors record similar data.

5. CONCLUSION:
The approach could have been applied to all other readings taken by the sensor such humidity, lighting, voltage level of sensor. The clustering can be applied considered all the readings taken as a vector and find the structure involved.

BIBLIOGRAPHY:
[1] *Distributed Regression: an Efficient Framework for Modeling Sensor Network Data* by *Carlos Guestrin,Peter Bodik, Romain Thibaux, Mark Paskin, Samuel Madden*
http://www.cs.cmu.edu/~guestrin/Publications/IPSN2004/ipsn2004.pdf