Neural Networks and Deep Learning (2024)

Assignment (1):

Clustering with SOM, Classification with SLFN

Shiraz University

Due date: 24th Faravardin 23:59 [12/April (2024)]

1 Part I: SOM Clustering

1.1 Winner-Takes-All (WTA) approach

Dataset: DOWNLOAD LINK

Recently, we have witnessed the rapid spread of the Coronavirus around the globe. Due to the importance of finding ways to control the spread of the virus, many researches have been put forward on.

In this part, you are going to do an analysis of the spatial evolution of coronavirus pandemic around the world by using a particular type of unsupervised neural network, SOM. Based on the clustering abilities of self-organizing maps you'll be able to spatially group together countries that are similar according to their coronavirus cases, in this way we will be able to analyze which countries are behaving similarly and thus can benefit by using similar strategies in dealing with the spread of the virus.

For each provided dataset, build a linear SOM with one neuron for each cluster. You have to assign the countries to a total of 4 clusters which indicate four labels "Low", "Medium", "High", and "Very High". Your final clusters contain the names of each country. Assign the mentioned labels to each cluster based on the average of total cases within the countries of that cluster.

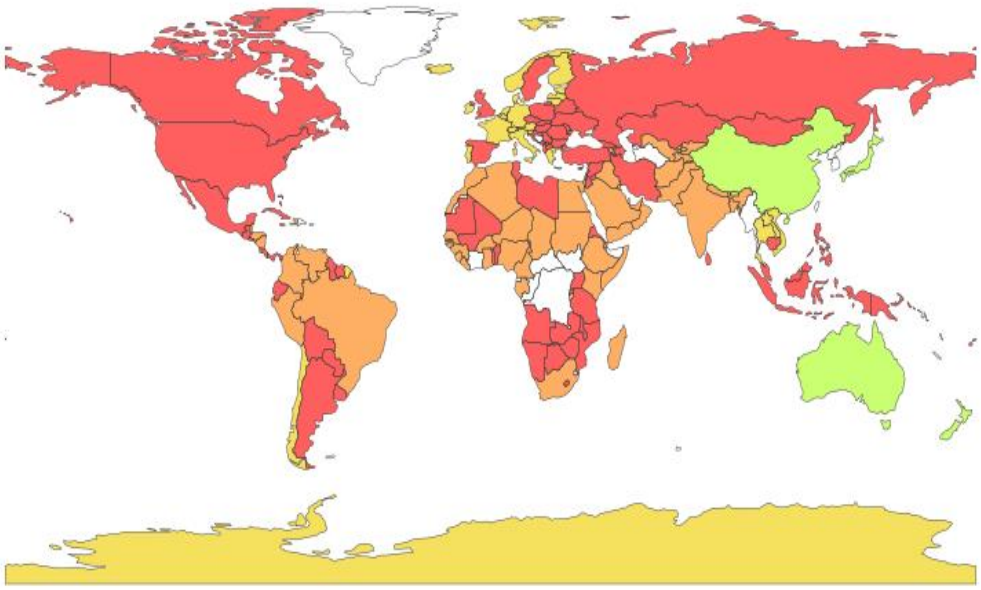

Figure 1: Plot of the world map showing different colour for each country belonging to a cluster. White coloured countries indicate missing country data from dataset.

Your dataset includes more than one instance for some countries representing their states. In such cases, you must find the average values of all its states. You must also drop longitude and latitude features which are irrelevant to our problem.

Be aware to normalize/standardize features if necessary. Report your final clusters in form of table and show the average number of cases in each country alongside them.

To visualize your results, you must plot the map of countries, with each country having a color representing is belonging cluster.

1.2 On-Center, Off-Surround approach

Dataset: DOWNLOAD LINK

Split the provided dataset into test and training sets. Using all training set data, build a rectangular SOM with 30×30 and 20×20 neurons (with neighborhood topology). After training the model with enough epochs, plot the winner map. In order to obtain the winner-map you have to try these two approaches:

1. **Voting approach:** The label of a neuron will be the label with the maximum number of assignments to that neuron.
2. **K-means**: You have to apply K-means algorithm on the final weights of obtained map to assign a label for each neuron.

Test your model with test dataset and print the true labels of instances in each cluster. Do the same thing for the clusters obtained with both voting and K-means approaches.

Evaluate you clustering (in both approaches) with different clustering evaluation metrics like Davies-Bouldin score and Normalized Mutual Information.

2 Part II: SLFN Classification

Dataset: DOWNLOAD LINK

In real use cases, training deep learning models are not always easy and might require more hardware and resources. For this part you must implement a Single Layer Feed-forward Network (SLFN) from scratch. SLFN is a simple model but to simulate the problem of lack of sufficient hardware you'll have a limit on the maximum number of epochs you can use to train your model.

Split the provided dataset into test and training sets. You can only train your SLFN model for the maximum of 5 epochs! With this limitation, your model might not perform very well on the unseen data. You must think of a way to overcome this issue. You must also be able to justify your solution in your reports.

The dataset for this part is same as the one provided in section 1.2 with an additional 'test.csv' file. The necessary information about this file will be announced later. You don't need to use in your training process. Finally, evaluate your model using different metrics like accuracy, precision, f1-score, recall and report them.

Notes:

- Allowed programming languages: Python, MATLAB
- You can use pre-defined libraries for this assignment, except for SLFN implementation.
- Any sign of cheating would result in a zero grade for this assignment.
- You should upload your submissions at:
  https://quera.org/course/add_to_course/course/16595/
  All of the files should be in a ZIP file named in this format:
  "FirstNameFamilyName-SudentNumber.zip"
  Ex: "AmirZamani-4023040.zip"
- Your reports should be in a PDF file including: key points of your implementation, explanation of your chosen approach, reports of your final results and answers of assignment questions (if given).