Combination of Vector Quantization and Visualization

Olga Kurasova\textsuperscript{1,2} and Alma Molytė\textsuperscript{1}

\textsuperscript{1} Institute of Mathematics and Informatics, Akademijos str., 08663 Vilnius, Lithuania
kurasova@ktl.mii.lt, alma.molyte@gmail.com
\textsuperscript{2} Vilnius Pedagogical University, Studentu str., 08106 Vilnius, Lithuania

Abstract. In this paper, we present a comparative analysis of a combination of two vector quantization methods (self-organizing map and neural gas), based on a neural network and multidimensional scaling that is used for visualization of codebook vectors obtained by vector quantization methods. The dependence of computing time on the number of neurons, the ratio between the number of neuron-winners and that of all neurons, quantization and mapping qualities, and preserving of a data structure in the mapping image are investigated.

1 Introduction

Any set of objects may often be characterized by some features $x_1, x_2, \ldots, x_n$. A combination of values of all features characterizes a particular object $X_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ from the whole set $X = \{X_1, X_2, \ldots, X_m\}$, where $n$ is the number of features and $m$ is the number of analyzed objects. As the number of features is often more than two, we deal with multidimensional data. So $X_1, X_2, \ldots, X_m$ are $n$-dimensional vectors. Often they are interpreted as points in the $n$-dimensional space $\mathbb{R}^n$, where $n$ defines the dimensionality of space. In fact, we have a matrix of numerical data $X = \{X_1, X_2, \ldots, X_m\} = \{x_{ij}, i = 1, \ldots, m, j = 1, \ldots, n\}$. The rows of this matrix are vectors $X_i = (x_{i1}, x_{i2}, \ldots, x_{in}), i = 1, \ldots, m$, where $x_{ij}$ is the $j$th component of the $i$th vector. The data matrix can be analyzed by various statistical methods. Especially when the amount of data is huge, the statistical methods are often not sufficient. In order to get more knowledge from the analyzed data, it is necessary to use data mining methods. A lot of data mining methods are developed for multidimensional data analysis: classification, clustering, visualization, etc.

In this paper, we analyze clustering and visualization of multidimensional data. Clustering is useful for reducing the amount of data. Vector quantization is performed in clustering methods. Visual data mining aims at integrating a human in the data analysis process, applying human perceptual capabilities to the analysis of large data sets available in today's computer systems. When combining clustering and visualization it is possible to get more knowledge than by the methods used individually.
2 Vector Quantization and Visualization

Vector quantization is a classical signal-approximation method that usually forms a quantized approximation to the distribution of the input data vectors $X_l \in \mathbb{R}^n, l = 1, \ldots, m$, using a finite number of so-called codebook vectors $M_i \in \mathbb{R}^n, i = 1, \ldots, N$. Once the codebook is chosen, the approximation of $X_l, l = 1, \ldots, m$, means finding the codebook vector $M_i$ closest to $X_l$, usually in the Euclidean metric $[1]$. Vector quantization is used for data compression, missing data correction, etc. It can be used for data clustering, too. In that case, the codebook vectors are representatives of clusters. Some methods for vector quantization are based on neural networks: self-organizing map (SOM) $[1]$, neural gas (NG) $[2]$, learning vector quantization $[1]$. Here the neurons correspond to the codebook vectors.

The self-organizing map (SOM) is a class of neural networks that are trained in an unsupervised manner using a competitive learning $[1]$. It is a well-known method for vector quantization. Moreover, the SOM is used for mapping a high-dimensional space onto a low-dimensional one. The neural gas is a biologically inspired adaptive algorithm, proposed in $[2]$. It sorts the input signals according to how far away they are. The learning vector quantization is a supervised method for data classification. It is not analyzed in this paper.

It is purposeful to visualize the codebook vectors obtained after quantization in order to get more knowledge from the analyzed data set. A large class of methods has been developed for multidimensional data visualization $[3]$, $[4]$. The visual presentation of the data allowed us to see the data structure, clusters, outliers, and other properties of multidimensional data. In this paper, one of the most popular methods of visualization, i.e., multidimensional scaling $[5]$ is used for visualization of codebook vectors.

2.1 Self-Organizing Map and Neural Gas

An array of vectors (codebook) $M$ is formed in both the neural gas network and the self-organizing map. Here the codebook vectors are often called neurons. The array $M$ is one-dimensional in the neural gas, $M = \{M_1, M_2, \ldots, M_N\}$, where $M_i = (m_{i1}, m_{i2}, \ldots, m_{in}), i = 1, \ldots, N$, $N$ is the number of codebook vectors. A self-organizing map is a two-dimensional grid. Usually, the neurons are connected to each other via a rectangular or hexagonal topology. The rectangular SOM is a two-dimensional array of neurons $M = \{M_{ij}, i = 1, \ldots, k_x, j = 1, \ldots, k_y\}$, where $M_{ij} = (m_{ij1}, m_{ij2}, \ldots, m_{ijn}), k_x$ is the number of rows, $k_y$ is the number of columns, and the total number of neurons is $N = k_x \times k_y$. The goal of these quantization methods is to change the values of codebook vectors (neurons) so that they would represent the properties of the vectors $X_l, l = 1, \ldots, m$ analyzed. At the end of learning, the codebook vectors become quantized vectors of the vectors $X_l$.

The neural gas algorithm is as follows:

1. Selecting initial values:
   - the number of codebook vectors (neurons) $N$;
   - values of the parameters $\lambda_i, \lambda_f, E_i, E_f$ used in the learning rule;