Decentralized Estimation Using Learning Vector Quantization

Mihajlo Grbovic Slobodan Vucetic*

Department of Computer and Information Sciences, Temple University

A decentralized estimation system consists of n distributed data sources $S_1 \ldots S_n$ and a fusion center. The data sources produce multivariate random vectors $X_1 \ldots X_n$ that are transmitted to the fusion center in the form of messages $Z_1 \ldots Z_n$, $Z_i = \alpha_i(X_i)$. Due to communication constraints, Z_i is a discrete variable with cardinality M_i represented as an integer from a set $\{1 \ldots M_i\}$. At the fusion center, the goal is to estimate the conditional expectation of unobserved variable Y, $E(Y \mid x_1 \ldots x_n)$, by fusion function $h(z_1 \ldots z_n)$. The challenge is to find quantization functions $\alpha_1 \ldots \alpha_n$ and fusion function h such that the estimation error is minimized under given communication constraints.

We studied a decentralized system where each sensor communicates only with the fusion center and there is no feedback to the sensors. It is assumed that a sample of size N from the underlying distribution, $D = \{(x_{1i} \ldots x_{ni}, y_i), i = 1 \ldots N\}$, is available for the design of the decentralized system. Our approach follows the iterative procedure of the generalized Lloyd's algorithm: (1) the quantizer for a single sensor is determined assuming the quantizers of the remaining sensors and the fusion function are known, (2) the fusion function is determined assuming all the quantizers are known.

For decoder design, the fusion function h is considered as the lookup table. Assuming $n = 2$, each of its elements is estimated by averaging the corresponding target values from the training data, $h(z_1,z_2) = \mathit{average}\ \{y_i: \alpha_1(x_{1i}) = z_1 \wedge \alpha_2(x_{2i}) = z_2\}$. For design of quantizer α_1, given quantizer α_2 and the fusion function h, the challenge is in partitioning the domain of X_1 into M_1 regions (one for each codeword) such that the estimation error is minimized. We adopted the multi-prototype approach where a set of B_1 ($\geq M_1$) prototype vectors are available for partitioning, each prototype is assigned to one of the M_1 codewords, and each observation is assigned to the codeword of its nearest prototype.

To determine prototype locations, we proposed a modification to the Learning Vector Quantization (LVQ) classification algorithm that updates locations of prototypes each time a data point is misclassified. Our Distortion-Sensitive LVQ (DS-LVQ) treats the decoder design as a soft classification problem. It calculates squared errors of assigning ith training observation x_{1i} to j-th codeword, as $e_{ij} = (y_i - h(j, \alpha_2(x_{2i})))^2$ and uses them to drive the updates of prototype locations.

The experiments were performed on single- and 2-source 2-dimensional problems and on 4-source 1 to 3 dimensional problems. We used 2 different training set sizes and several levels of noise. The proposed algorithm was compared to the regression tree algorithm and the regular LVQ algorithm. The results showed that the proposed algorithm is robust to noise and that is achieves better estimation than the alternatives.

* This work is supported by the U.S. National Science Foundation Grant IIS-0546155. Contact: vucetic@ist.temple.edu