# Learning Multiple Latent Variables with Self-Organizing Maps

Lili Zhang
*Rice Quantum Institute and*
*Department of Electrical & Computer Engineering*
*6100 Main Street, Rice University*
*Houston, TX 77005 USA*
*Email: llzhang@rice.edu*

Erzsébet Merényi
*Department of Electrical & Computer Engineering*
*6100 Main Street, Rice University*
*Houston, TX 77005 USA*
*Email: erzsebet@rice.edu*

***Abstract*—Inference of latent variables from complicated data is one important problem in data mining. The high dimensionality and high complexity of real world data often make accurate inference difficult. We approach this challenge with a neural architecture we call Conjoined Twins, which is a two-layer feedforward network with a Self-Organizing Map (SOM) as its hidden layer. Its output layer can preferentially use different numbers ($k$) of SOM winners for the inference of different latent variables. We introduced this architecture in [1], [2]. In this paper we propose an automated procedure for the customization of $k$ and demonstrate the effectiveness of the method by the inference of two physical parameters of icy planetary surfaces from spectroscopic data.**

***Keywords*-Self-Organizing Map; latent variable; planetary spectra**

## I. Inference of latent variables from high-dimensional observable data

Data collected to characterize a real world process or problem are usually high-dimensional, providing a wealth of information that can potentially be exploited by the observer. The elements of the high-dimensional data vectors are called *observable variables*. *Latent variables*, which are impossible or hard to observe directly, have to be inferred from the observable data. However, the high dimensionality as well as the often accompanying high complexity of the data make it difficult to extract the latent variables. Due to the "curse of dimensionality", parametric modeling becomes problematic because the requirement for large number of data samples in the estimation of parameters is often unmet. Motivated by the idea that the data samples, although high-dimensional, can lie on a low-dimensional submanifold, another strategy is to embed the data in a low-dimensional space (dimension reduction), as in a number of manifold learning algorithms (e.g., [3], [4], [5]). The resulting low-dimensional representation is intended to factorize the latent variables, or the degrees of freedom, of the data set. A classical approach, principle component analysis, works well for linear submanifolds, but suffers when nonlinearities exist. Several nonlinear approaches, such as Isomap [4] and locally linear embedding [5], have been successful in applications where there is a small number of latent variables (e.g., pose of face and illumination direction in data sets of face images [6]), but might lose effectiveness when the number of latent variables increases. In addition, to separate all latent variables, a necessary step for these algorithms is to estimate the intrinsic dimension (the total number of latent variables), which is a nontrivial task. Moreover, latent variables that induce relatively small variations in the data can be lost in such dimension reduction. Our approach to this problem is through Self-Organizing Maps (SOM) [7], which preserve the topology of the data in a low-dimensional representation without reducing the data dimensionality. SOM is an adaptive vector quantizer, which spreads vector quantization prototypes optimally throughout the manifold to capture the structure, and at the same time organizes the prototypes in a low-dimensional lattice. Although the influences of different latent variables can be convoluted in the SOM representation, we can disentangle those influences by using linear combinations of different numbers of SOM outputs for the recovery of different variables. To do this, we use a neural architecture we call Conjoined Twins, introduced in [1], [2], the output layer of which is trained to infer latent variables with the help of the SOM. In this paper we propose an automated approach for the most essential step in the Conjoined Twins, namely the customization of the number of SOM outputs to use for each latent variable.

## II. Conjoined Twins – a $k$-Winners-Take-All ($k$WTA) SOM-hybrid neural architecture

Before elaborating on the customization procedure (Section III), which is the focus of this paper, we briefly review the motivation and the concept of the Conjoined Twins in this section. The Conjoined Twins was developed from an SOM-hybrid neural architecture, when we needed to infer two highly nonlinearly dependent variables for a planetary surface characterization problem. The SOM-hybrid neural architecture is a two-layer fully connected supervised feed-forward network, with a 2-dimensional SOM as the hidden layer (Fig. 1). The customary handling of the SOM outputs, or the Winner-Takes-All (WTA) mode, was beneficial for the inference of one variable, but not for the other [1], [2], therefore we generalized the use of the SOM such that any number ($k$) of SOM winners could contribute to the

supervised learning in the output layer (section II-B). The idea of the Conjoined Twins was to combine the use of different numbers ($k$) of SOM winners for different latent variables in one architecture (section II-C).

### A. The SOM-hybrid neural architecture

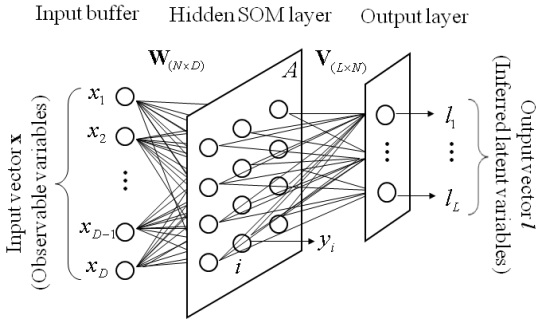


Figure 1. The SOM-hybrid neural architecture. Each neuron $i$, in the SOM lattice $A$ of $N$ neurons, is connected to the input buffer with a $D$-element prototype $\mathbf{w}_i$ (the $i$th row vector of the $N \times D$ matrix $\mathbf{W}$). An $L \times N$ weight matrix $\mathbf{V}$ connects the output layer to the SOM.

The SOM-hybrid neural architecture, as shown in Fig. 1, takes an input vector $\mathbf{x} = [x_1, x_2, ..., x_D]^T$ randomly from the $D$-dimensional data set in each learning step. In the first, unsupervised, learning phase of this machine, the SOM iteratively adjusts its $N$ prototypes, $\mathbf{w}_i$, according to the SOM algorithm [7] while the output layer is idle. We use the Conscience SOM variant [8] to achieve equal winning probabilities across all SOM neurons, producing a more faithful *pdf* matching than the original Kohonen's algorithm. Briefly, for each input vector $\mathbf{x}$, an SOM winner or the best matching unit (BMU) $c$ is determined as

$$c = \arg\min_j (\| \mathbf{w}_j - \mathbf{x} \|^2 - b_j) \quad \forall j \in A \tag{1}$$

The bias $b_j$ is computed from the winning frequency $p_j$, of SOM neuron $j$, as

$$b_j = \gamma(t) \times ((N \times p_j) - 1) \tag{2}$$

where $\gamma$ is a parameter. All prototypes $\mathbf{w}_j$ are then updated:

$$\mathbf{w}_j^{new} = \mathbf{w}_j^{old} + \alpha(t) h_{c,j}(t)(\mathbf{x} - \mathbf{w}_j^{old}) \tag{3}$$

$\alpha$ is the learning rate. $h_{c,j}(t)$ is a neighborhood function. In the Conscience algorithm $h_{c,j}(t)$ can be fixed and of small size (e.g., the immediate neighbors in a diamond or square configuration), instead of the commonly used large neighborhood (e.g., Gaussian) that has to decrease with time. A topologically ordered mapping of the input data forms during this learning phase, which reflects the structure of the high-dimensional manifold. Upon the convergence of the SOM, the output layer is turned on and a second, supervised, learning phase starts. Each neuron $p$ in the output layer combines the SOM outputs into a weighted sum

$$l_p = \sum_{i \in A} v_{pi} y_i \qquad p = 1, 2, ..., L \tag{4}$$

and minimizes the total squared error in the outputs $l_p$ by iteratively adjusting $\mathbf{V}$ according to the delta rule [9]. $v_{pi}$ is the element of $\mathbf{V}$ in the $p$th row and $i$th column. When this network is used to infer latent variables, the output vector, $\boldsymbol{l}$ $(= [l_1, l_2, ..., l_L]^T)$, yields the inferred variable values.

This supervised architecture is suitable for the analysis of high-dimensional data mainly for two reasons. One is the ease and economy of the SOM in the handling of high-dimensional data, compared to other more frequently used approaches. The other is the ability of the SOM to distinguish the subtle differences between high-dimensional feature vectors, which has generally helped achieve good prediction accuracies [10], [11].

The customary way of passing the SOM's knowledge to the output layer is the winner-takes-all (WTA) mode, which assigns an output value of 1 to the BMU $c$ and 0 to the rest.

$$y_i = \begin{cases} 1 & i = c \\ 0 & i \neq c \end{cases} \tag{5}$$

By this, the right side of eq. 4 is reduced to one term.

$$l_p = v_{pc} \qquad p = 1, 2, ..., L \tag{6}$$

With a single term left in the weighted sum the network will be unable to distinguish between the data samples that map to the same SOM neuron and will yield the same inferred value for these samples. In problems where the number of different values a latent variable can take is much smaller than $N$, the WTA mode can work successfully in differentiating these various values. However, when the latent variable is continuous, i.e., the number of possible values is much larger than $N$, the resolution of the inferred values is severely restricted by the WTA mode, which may prevent high inference accuracies.

### B. Exploitation of the SOM's knowledge with kWTA

To relieve the above limitation in the inference resolution caused by the WTA mode, we allow multiple ($k$) SOM outputs to be nonzero in eq. 4 ($k$-winners-take-all or $k$WTA). This can be justified by the SOM algorithm: the prototypes within the lattice neighborhood of the BMU learn concurrently from the same input vector (eq. 3). The memory of the data is stored not only in the BMUs but also in their neighbors. Observing that the activation levels of the SOM neurons are inversely proportional to the distance $d_i$ between the prototype $\mathbf{w}_i$ and the input vector $\mathbf{x}$, a natural choice for $k$WTA is to use the top $k$ SOM outputs:

$$y_i = \begin{cases} 1 & i = i_1 (= c) \\ \frac{d_1}{d_1 + d_i} & i = i_2, i_3, ..., i_k \\ 0 & i \neq i_1, i_2, ..., i_k \end{cases} \tag{7}$$

We then normalize $y_i$ to make the total output from the SOM sum up to 1, which is consistent with the WTA mode.

$$y_{i_q} = \frac{y_{i_q}}{\sum_{q=1}^{k} y_{i_q}} \quad q = 1, 2, ..., k \tag{8}$$

Each output is now expressed as a linear combination of $k$ nonzero SOM outputs.

$$l_p = \sum_{q=1}^{k} v_{pi_q} y_{i_q} \quad p = 1, 2, ..., L \tag{9}$$

This can help improve the inference capability for continuous latent variables. WTA is obviously a special case of $k$WTA ($k = 1$). NeuralWare's implementation in Neural Works Professional II/Plus [12] provides the special cases of $k = 1$ and $k = 3$.

### C. Conjoined Twins architecture

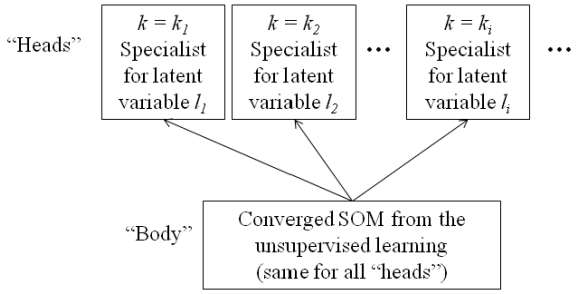


Figure 2. Conceptual diagram of a Conjoined Twins architecture.

Each latent variable can have a different effect on the manifold structure. Consequently, their recovery may be best done from mixtures of different numbers of SOM prototypes. The idea of the Conjoined Twins we proposed in [1], [2] is to combine the use of customized mixtures of SOM prototypes in a single architecture, as shown in Fig. 2. It has multiple copies of the output layer in Fig. 1, which we call "heads". All "heads" rely on the same "body" of knowledge, the learned SOM, but each draws from a different $k_i$ number of SOM winners in the weighted sum (eq. 9) for best learning of $l_i$. This architecture provides a solution for the inference of multiple latent variables with minimal increase in computational cost compared to the SOM-hybrid machine in Fig. 1. As we showed in [1], [2] for the inference of two latent variables, the essential step, i.e., the customization of $k$ for each latent variable, was accomplished through manual evaluation of the SOM. The focus of this paper is to propose a procedure to automate this customization step.

## III. Automatic customization of $k$

Two questions to answer for the determination of $k$ are: how many SOM winners are *sufficient* to represent the information in a data sample, i.e., what is the collective upper limit, $K$, of $k$ for all latent variables, and what is the *necessary* smallest $k_i \leq K$ for the supervised learning of $l_i$. We propose a two-step procedure accordingly.

### A. Determination of the upper limit of $k$

We determine the upper limit of $k$ from the relative *importance* of SOM winners. For any data sample, the BMU (or the first ranking winner) is the most important one, containing the most information about that given sample. The importance of the other winners can be evaluated by their similarities to the BMU. As was shown in [15], the so-called induced Delaunay graph could represent this similarity relationship more faithfully than the customary Euclidean metric when discontinuities and nonlinearities exist in the manifold structure. According to a procedure proposed by Martinetz and Schulten [15], an edge, or a *connection*, in the induced Delaunay graph is constructed between two SOM prototypes that form a pair of BMU and second BMU for at least one data sample. These two prototypes are *Voronoi neighbors*. An example of an induced Delaunay graph is given in Fig. 3, left, for the two-dimensional "clown" data set from [13], the structure of which mimics a clown's face. The induced Delaunay graph (black lines) delineates the discontinuities between different parts, such as between the eyes and the nose and between the mouth and the body, whereas the regular Delaunay graph (grey lines) does not show the same separations. However, with the induced Delaunay graph, noisy data can still easily obscure discontinuities because one data sample is enough to establish a connection. To distinguish the important connections from the unimportant ones, a *connection strength*, $CONN(i, j)$, was defined between prototypes $\mathbf{w}_i$ and $\mathbf{w}_j$ as the number of data samples that choose these two prototypes as the BMU and the second BMU [14]. The $CONN$ matrix is then the induced Delaunay graph whose edges are weighted by the connection strengths. It reflects the anisotropic data distribution in the Voronoi cells of the prototypes (Fig. 3, right) and uses this information to interpret the similarity relationships between the prototypes. Using the $CONN$, the discontinuities obscured by noise can emerge. For example, in Fig. 3, middle, the separations between the three subclusters in the left eye (right on the figure) become visible. The advantage of the $CONN$ in unravelling detailed structure in a manifold makes it a suitable basis for the determination of the upper limit $K$. A loose upper limit can simply be determined as $K = m + 1$, where $m$ is the maximum number of Voronoi neighbors to any prototype, because only the Voronoi neighbors share the same information about a given datum as the BMU. Usually $m$ is already much smaller than the total number of SOM prototypes, $N$, but we can further tighten this limit by examining the strengths of the connections to these $m$ neighbors. For each prototype, we rank all of its Voronoi neighbors according to their connection strengths. For example, in Fig. 3, right, among the four neighbors of prototype P1, the first ranking (the most similar) to the last ranking (the least similar) neighbors are P2, P3, P4 and P5, in decreasing order of connection strengths. By computing the average connection strength $s_i$ to the $i$th ranking neighbors across all SOM prototypes

$$s_i = \frac{1}{n_i} \sum_{\substack{p,q \in A \ \wedge \ \mathbf{w}_q \text{ is the} \\ i\text{th Voronoi neighbor of } \mathbf{w}_p.}} CONN(p, q) \tag{10}$$

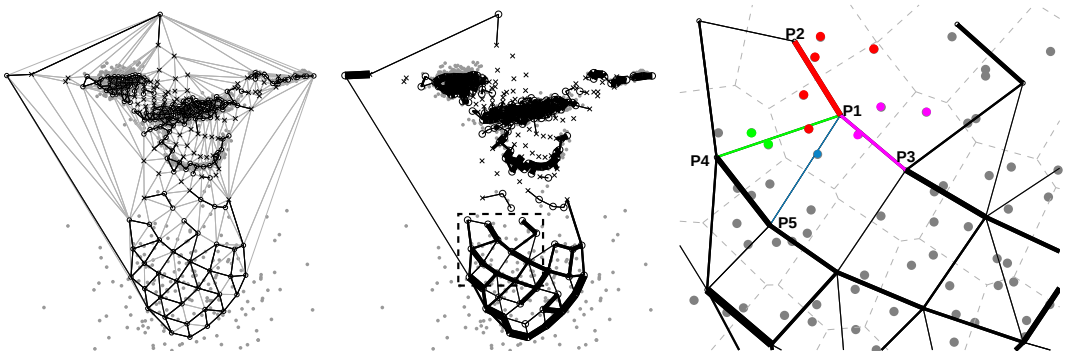


Figure 3. Illustration of the Delaunay graph, the induced Delaunay graph and the $CONN$ matrix with the two-dimensional "clown" data set from [13], where the manifold structure mimics a clown's face (two eyes, a nose, a mouth and a body). Open circles and crosses represent the SOM prototypes with and without samples mapped to them, respectively. Grey dots are data samples. Left and middle figures are from [14]. **Left:** The induced Delaunay graph (black lines) highlights most of the discontinuities in the manifold structure, which are hardly seen from the Delaunay graph (gray lines). **Middle:** The $CONN$ matrix, which weighs each connection in the induced Delaunay graph with its connection strength, makes more detailed structures visible, such as the three subclusters in the left eye. Part of the body, in the dashed square, is magnified on the right. **Right:** Magnified detail from the upper left part of the clown's body, as an example of how the $CONN$ matrix reflects the anisotropic data distribution in the Voronoi cells (dashed lines). The 4 connections from prototype P1 to P2, P3, P4 and P5 have strengths 5, 3, 2 and 1, respectively. To make this easy to see, we colored the data samples that contribute to each of the 4 connections, the same as their corresponding connections. According to the connection strengths, P2, P3, P4 and P5 can be ranked as the first (the most similar) to the fourth ranking (the least similar) neighbors of P1.

we can quantify the importance of neighbors of each rank. $n_i$ $(i = 1, 2, ..., m)$ is the total number of $i$th ranking neighbors in the SOM. Since $s_i$ is dependent on several factors, e.g., the size of the SOM, the size of the data set and how the data samples are distributed across the Voronoi cells, we normalize $s_i$ by the average strength of all connections in the SOM, $\overline{s}$, as in eq. 11, such that a consistent thresholding can be proposed for the determination of $K$ for different SOMs learned with different data sets.

$$norm\_s_i = \frac{s_i}{\overline{s}} \tag{11}$$

$$\overline{s} = \frac{\sum_{p,q \in A} CONN(p,q)}{\sum_{i=1}^{m} n_i} = \frac{P}{\sum_{i=1}^{m} n_i} \tag{12}$$

where $P$ is the total number of data samples. We can then apply a threshold $\mu_1$ (a user-chosen parameter) to $norm\_s_i$ to find the number of important neighbors, $i_{max}$.

$$i_{max} = \max\{i : norm\_s_i > \mu_1\} \tag{13}$$

$K$ is then determined as $i_{max} + 1$ (including the BMU).

There is one situation where $norm\_s_i$ may be insufficient for selecting $K$: when the connections to $i$th ranking neighbors are weak (small $s_i$), but $n_i$ is large, the neighbors of rank $i$ may still be useful in the supervised learning. In view of this, it is better to consider the combined effect of $s_i$ and $n_i$ in the thresholding. For this purpose, we propose an alternative, thresholding on the percentage of data samples, $\%data_i$, involved in the connections of each rank $i$ with a user-chosen parameter $\mu_2$:

$$i_{max} = \max\{i : \%data_i > \mu_2\} \tag{14}$$

$$\%data_i = \frac{s_i \times n_i}{2P} \times 100\% \tag{15}$$

$s_i \times n_i$ can be interpreted as the accumulated strength in the connections to all $i$th ranking neighbors. Normalized by $2P$, it shows the importance of these neighbors by the percentage of the total connection strength involved. We will use this thresholding with $\mu_2 = 1\%$ for the automatic determination of $K$ in this paper, assuming that connections contributed by less than $1\%$ of data samples are negligible.

### B. Searching for the best $k$ below the upper limit $K$

$K$ has been determined in the previous step as the maximum of the number of SOM winners that potentially carry information about each data sample. However, information about a specific latent variable $l_i$ can be contained in a smaller number, $k_i \leq K$, of winners. Therefore, the second step of the customization is to search for $k_i$, the *necessary* number of SOM winners, for each latent variable $l_i$, below the upper limit $K$. Since the search range of $k_i$ has already been narrowed by $K$, we perform an exhaustive search by repeating the supervised phase of the training $K$ times with $k = 1, 2, ..., K$ and selecting $k_i$ with which we obtain the highest inference accuracy for $l_i$.

After this, we can "mount" the "heads" of the Conjoined Twins (Fig. 2) by attaching the weight matrix $\mathbf{V}$ trained with $k = k_i$ to the head that specializes on the inference of $l_i$.

## IV. Application to Inference of Two Physical Parameters from Near-Infrared Spectra

We apply the Conjoined Twins to the inference of temperature and grain size from Near-Infrared high-dimensional spectral data that will be collected from icy surfaces in the Pluto-Charon system. Temperature and grain size are two latent variables that influence the remote sensing spectra, which are often used to recover these physical parameters for extended surface areas. Both temperature and grain size have global influence on the spectral shapes, but in different ways. As seen from two representative sets of sample spectra

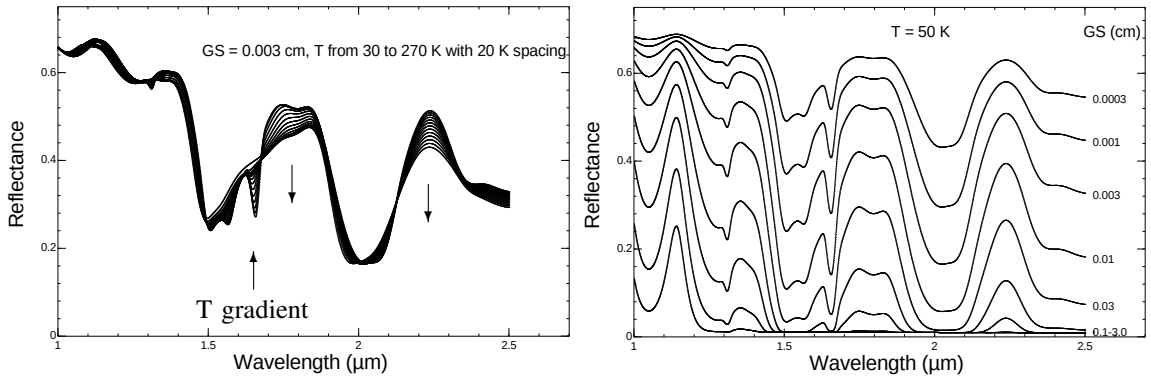


Figure 4. Variations in spectra of $H_2O$ ice as a function of temperature (T) and grain size (GS). Figures are from [2]. **Left**: Spectra as a function of 13 different temperatures for one fixed grain size, 0.003 cm. **Right**: Spectra as a function of 9 different grain sizes at a fixed temperature, 50 K.

of $H_2O$ ice in Fig. 4, both parameters deepen the absorptions, e.g., at 1.3 $\mu$m and 1.65 $\mu$m, but the band depth is a nonmonotonic function of grain size whereas it changes monotonically with temperature, at a given wavelength. In addition, temperature has a much smaller effect on the spectral brightness than grain size. The subtle change in spectral shape caused by temperature makes the discrimination between spectra with different temperatures, as well as the accurate inference of temperatures, difficult. The data we use here are synthetic spectra of $H_2O$ ice generated through a radiative transfer code [16], [17] on a parameter grid of 126 temperatures, with 2 K spacing between 20 and 270 K, and 81 grain sizes logarithmically spaced from 0.0003 to 3.0 cm. Each data vector is a 230-dimensional spectrum. We use a 20×20 and a 40×40 SOM to learn the same data set and investigate how the customization of $k$ works differently in these two cases. The prediction accuracies are computed as the percentage of test data samples that are predicted with less than 5% error.

In the unsupervised learning phase, the whole data set is used, whereas in the supervised phase the data set is randomly split into a training and a test set with a 9:1 ratio. When the unsupervised learning ends, $K$ is automatically determined as 6 and 4 for the 20×20 and the 40×40 SOMs, respectively, from the statistics computed from the $CONN$ matrix (Table I). From the supervised training runs repeated with different values of $k$ (Table II), the best $k$ for each latent variable is also automatically identified from the highest inference accuracies (in bold face). The customization results are: $k = 3$ and $k = 4$ with the 20×20 SOM, and $k = 1$ and $k = 4$ with the 40×40 SOM, for the inference of grain size and temperature, respectively. With the 20×20 SOM, we achieve 77.9% and 53.9%, and with the 40×40 SOM, we achieve 97.9% and 81.3% prediction accuracies for grain size and temperature. When the number of SOM winners used in the learning is less than the customized best $k$, the accuracy is lower than the best result because of the loss of information contained in the excluded winners; when the number of SOM winners is larger than the best $k$, the poorer accuracy can result from the fact that we include more winners but we use the same number of training steps as with less than $k$ winners. The possibility of higher inference accuracies by including more than $k$ winners, with longer supervised training, will be investigated in future work.

Table I

STATISTICS OF THE CONNECTIONS TO VORONOI NEIGHBORS, FROM THE HIGHEST RANKING TO THE LOWEST, COMPUTED ACROSS ALL SOM PROTOTYPES FOR THE 20×20 AND THE 40×40 SOMS LEARNED WITH THE SPECTRAL DATA SET. THE $\%data_i$ VALUES ABOVE THE THRESHOLD $\mu_2$ (=1% IN THIS PAPER) ARE SHOWN IN BOLD FACE. THE UPPER LIMIT $K$ IS AUTOMATICALLY DETERMINED AS 6 AND 4 FOR THE 20×20 SOM AND THE 40×40 SOM, RESPECTIVELY.

| | | Neighbor ranking $i$ | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 20×20 SOM | $n_i$ | 380 | 371 | 287 | 158 | 61 | 15 |
| | $s_i$ | 27.51 | 16.01 | 9.57 | 6.20 | 4.11 | 2.73 |
| | $norm\_s_i$ | 1.71 | 1.00 | 0.60 | 0.39 | 0.26 | 0.17 |
| | $\%data_i$ | **51.22** | **29.10** | **13.45** | **4.80** | **1.23** | 0.20 |
| 40×40 SOM | $n_i$ | 1262 | 1109 | 363 | 112 | 15 | 1 |
| | $s_i$ | 9.98 | 5.90 | 2.88 | 1.81 | 1.27 | 1.00 |
| | $norm\_s_i$ | 1.40 | 0.83 | 0.40 | 0.25 | 0.18 | 0.14 |
| | $\%data_i$ | **61.73** | **32.06** | **5.11** | 0.99 | 0.09 | 0.00 |

Table II

PREDICTION ACCURACIES (%) FOR GRAIN SIZE (GS) AND TEMPERATURE (T), AFTER 1 MILLION SUPERVISED TRAINING STEPS, WITH THE 20×20 AND 40×40 SOMS IN THE $k$WTA MODE, EACH WITH A DIFFERENT $k$ BELOW THE UPPER LIMIT $K$. AN ADDITIONAL RESULT IS SHOWN FOR THE 40×40 SOM WITH $k = 5$ BECAUSE ITS $\%data_4$ (0.99%) IS EXTREMELY CLOSE TO THE THRESHOLD $\mu_2$ (1%). THE HIGHEST INFERENCE ACCURACIES (IN BOLD FACE) INDICATE THE BEST CHOICES OF $k$. RESULTS ARE FROM SINGLE RUNS FOR REASONS OF TIME LIMITATIONS. FURTHER JACK-KNIFE RUNS ARE IN PROGRESS.

| | | $k$ | | | | | |
|---|---|---|---|---|---|---|---|
| SOM size | | 1 | 2 | 3 | 4 | 5 | 6 |
| 20×20 | GS | 73.0 | 72.4 | **77.9** | 73.7 | 67.9 | 66.2 |
| | T | 32.9 | 49.7 | 53.7 | **53.9** | 52.5 | 53.0 |
| 40×40 | GS | **97.9** | 51.9 | 54.3 | 47.4 | 47.2 | – |
| | T | 61.2 | 71.2 | 77.3 | **81.3** | 80.6 | – |

The different customization results for the two SOMs and the significant improvements in the inference accuracies achieved with the larger SOM can be explained by the different representations of the two latent variables. In the 40×40 SOM, we observed that 81 grain size clusters separated from each other almost cleanly [1]. This enables a near perfect

inference accuracy with $k = 1$ because each individual neuron represents a single grain size. We also observed that in each grain size cluster the prototypes varied in an orderly fashion according to temperatures [1]. Unlike grain size, temperature cannot be uniquely represented by single neurons because there are more possible values (126) than the number of neurons in each grain size cluster ($\sim$19). Each prototype forms a mixture of spectra with different temperatures, and a specific temperature can best be recovered from the mixtures in the first 4 winners ($k = 4$). This difference in the dominance of the two physical parameters on the SOM clustering can be easily understood from their different effects on the spectral shape (Fig. 4). However, in the small SOM (20×20), we found that the grain size clusters did not separate clearly. Due to the reduced size of the SOM, each prototype is forced to represent not only a mixture of different temperatures, but also a mixture of different grain sizes. This explains why multiple winners ($k = 3$) work better for the inference of grain size than a single winner (77.9% v.s. 73.0%). The overall performance suggests that the variations in data caused by the two latent variables are insufficiently represented in the small SOM, due to which they cannot be inferred as accurately as with the large SOM. However, we should note that the increase in computational burden is significant with increasing SOM size, and should be considered in the tradeoff between computational cost and inference capability. In this case, we are satisfied with the the 40×40 SOM because the results obtained with it are scientifically useful [1], [2].

## V. Conclusion

We propose a principled customization procedure for the Conjoined Twins neural machine, to automate the determination of the best number of the SOM winners for the learning of each of several latent variables. We show the effectiveness of the approach through a data set with two latent variables here. Follow-up work will assess the effectiveness for three or more latent variables. We also note the possible limitation in the linear model (eq. 4) we use to retrieve information from the SOM, and that alternative nonlinear models could further improve the inference accuracies.

## Acknowledgment

We thank Dr. Eliot F. Young and Dr. William M. Grundy for graciously providing the spectral data set and for their valuable discussions on this work. This research was partially supported by grants NNG05GA63G and NNG05GA94G from the Applied Information Systems Research Program, NASA, Science Mission Directorate.

## References

[1] L. Zhang *et al.*, "An SOM-Hybrid Supervised Model for The Prediction of Underlying Physical Parameters from Near-Infrared Planetary Spectra," in *Proc. 7th WSOM Advances in Self-Organizing Maps*, ser. LNCS 5629, J. Príncipe and R. Miikkulainen, Eds., St. Augustine, FL, 2009, pp. 362–371.

[2] L. Zhang *et al.*, "Inference of surface parameters from near-infrared spectra of crystalline $h_2o$ ice with neural learning," *Publications of the Astronomical Society of the Pacific, under revision*, 2010.

[3] T. F. Cox and M. A. Cox, *Multidimensional scaling*. Chpman and Hall/CRC, 2001.

[4] J. B. Tenenbaum *et al.*, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[5] S. Roweis and L. Soul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[6] J. Lee and M. Verleysen, *Nonlinear Dimension Reduction*, ser. Information Science and Statisics. Springer, 2007.

[7] T. Kohonen, *Self-Organizing Maps*, 3rd ed. Berlin Heidelberg New York: Springer-Verlag, 2001.

[8] D. DeSieno, "Adding a Conscience to Competitive Learning," in *IEEE Int'l Conference on Neural Networks*, vol. 1, 1988, pp. 117–124.

[9] B. Widrow and F. W. Smith, *Computer and Information Sciences (COINS) Symposium Proceedings*. Washington DC: Spartan Books, 1964, ch. Pattern-Recognizing Control Systems, pp. 288–317.

[10] E. S. Howell *et al.*, "Classification of Asteroid Spectra Using a Neural Network," *Jour. Geophys. Res.*, vol. 99, no. E5, pp. 10,847–10,865, 1994.

[11] L. Rudd and E. Merényi, "Assessing Debris-Flow Potential by Using AVIRIS Imagery to Map Surface Materials and Stratigraphy in Cataract Canyon, Utah," in *Proc. 14th AVIRIS Earth Science and Applications Workshop*, R. Green, Ed., Pasadena, CA, 2005.

[12] NeuralWare Inc., *Neural Computing – A Technology Handbook for Professional II/PLUS and NeuralWorks Explorer*, NeuralWare Inc., Pittsburgh, 1997.

[13] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Transactions on Neural Networks*, vol. 11, pp. 586–600, 2000.

[14] K. Taşdemir and E. Merényi, "Exploiting the data topology in visualizing and clustering of Self-Organizing Maps." *IEEE Transactions on Neural Networks*, vol. 20, no. 4, pp. 549–562, 2009.

[15] T. Martinetz and K. Schulten, "Topology representing networks," *Neural Networks*, vol. 7, pp. 507–522, 1994.

[16] W. M. Grundy, Ph.D. dissertation, University of Arizona, 1995.

[17] W. M. Grundy and B. Schmitt, "The temperature-dependent near-infrared absorption spectrum of hexagonal $H_2O$ ice," *Jour. Geophys. Res.*, vol. 103, pp. 25 809–25 822, 1998.