Evaluating Self-Organizing Map Quality Measures as Convergence Criteria

Gregory Breard
Dept. of Computer Science and Statistics
University of Rhode Island
Kingston, RI 02881–2018
Email: gtbreard@my.uri.edu

Lutz Hamel
Dept. of Computer Science and Statistics
University of Rhode Island
Kingston, RI 02881–2018
Email: lutzhamel@uri.edu

***Abstract*—The self-organizing map is a type of artificial neural network that has applications in a variety of fields. The self-organizing map training algorithm uses unsupervised learning to produce a low-dimensional representation of high-dimensional data. The low-dimensionality of the resulting map allows for a graphical presentation that is easily interpreted. It is essential to evaluate the quality of the maps to ensure that these models are representative of the underlying data. Various measures have been developed to quantify a map's quality. Little work, however, has been done comparing these measures to one another. To that end, this paper evaluates quality measures as convergence criteria. This is achieved by examining the underlying structure of maps that are converged under different measures. Specifically, the clusters that exist in the maps after they are reported to have converged are compared with the clusters that exist in the input data. The quality measures studied are quantization error, topographic error, trustworthiness, neighborhood preservation, and population-based convergence.**

I. Introduction

The self-organizing map (SOM) is a type of artificial neural network that has applications in a variety of fields and disciplines. The self-organizing map training algorithm uses unsupervised learning, or sometimes called competitive learning, to produce a low-dimensional representation of high-dimensional data. This is done by "fitting" a grid of nodes to a data set over a fixed number of iterations. The low-dimensionality of the resulting map allows for an easily interpreted graphical presentation of the data. An example visualization of a SOM trained with the multivariate *Ecoli* data set [11] is shown in Figure 1. Note that the clusters are easily identifiable in the two-dimensional map despite the fact that the data has seven dimensions. Although this might appear to be a "good" model, visual inspection is not sufficient to determine the quality of the map As we will see in our discussion later, this is actually not a very good model. Recall that in an unsupervised learning setting cluster labels are typically not available for cluster quality assessment.

A variety of quality measures have been developed over the years that attempt to quantify how well the underlying data is represented by a map. Some work has been done comparing these quality measures [13], however, the focus has generally been on the size of the map, not the amount of training. Here we compare quality measures by evaluating them as convergence criteria. This is accomplished by examining the structure of maps that are converged under different measures. Specifically, the clusters that exist in the maps will be compared to the clusters that exist in the training data. The quality measures studied are quantization error [9], topographic error [8], trustworthiness, neighborhood preservation [17], and population-based convergence [2].

The remainder of the paper is organized as follows. Section II provides a brief overview of the SOM training algorithm. We introduce the quality measures in Section III. In Section IV we describe our experimental procedure and we discuss our data sets we used for this study in Section V. Our results are reported in Section VI and we conclude the paper with some final remarks in Section VII.

II. The Self-Organizing Map

The SOM is an artificial neural network developed by Teuvo Kohonen [9]. The training algorithm uses an unsupervised, iterative procedure to model an input space with a fixed lattice of nodes.

A high-level version of the SOM training algorithm is shown in Algorithm 1. The SOM can also be described in terms more typical of artificial neural networks [5]. Given X, a set of n k-dimensional input vectors $\mathbf{x}_i \in \mathbb{R}^k, i = 1, ..., n$. Let M be a 2-dimensional grid of m neurons with $m = x \times y$, where x and y are the dimensions of the grid. Each neuron in M has a weight vector $\mathbf{w}_j \in \mathbb{R}^k$ with index $j = 1, ..., m$. The following training steps are repeated for a given number of iterations.

- Select an input $\mathbf{x}_i \in X$. Use (1) to determine the index of the best-matching unit (BMU) b in M for $\mathbf{x}_i$.

$$b = argmin_j(\|w_j - x_i\|) \tag{1}$$

Algorithm 1 The SOM training algorithm.

```
Initialize map.
repeat
    for each training observation do
        Find the neuron that best matches the observation.
        Make that neuron look more like the observation.
        Smooth the immediate neighborhood of that neuron.
    end for
until done
return trained map.
```

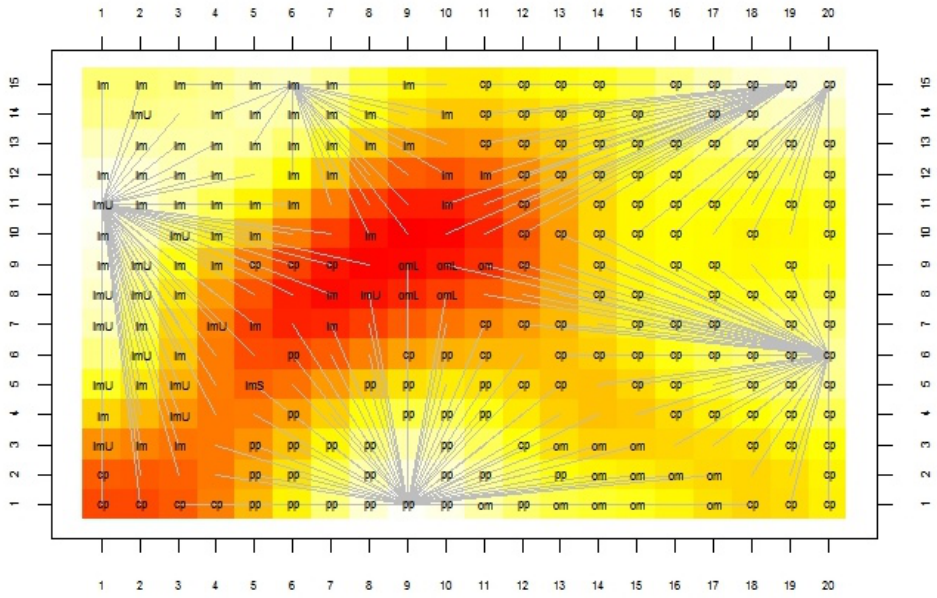

Fig. 1. Starburst visualization for *Ecoli* data set.

- The point $\mathbf{x}_i$ is used to update the BMU b and its neighboring nodes using (2) for all $j = 1, ..., m$, where α is the learning rate, r is the current neighborhood radius, $h(b, j, r)$ is the loss function, and $\delta_i = \mathbf{w}_j - \mathbf{x}_i$.

$$\mathbf{w}_j \leftarrow \mathbf{w}_j + \alpha\delta_i h(b, j, r) \tag{2}$$

The loss function $h(b, j, r)$ is defined as,

$$h(b, j, r) = \begin{cases} 1 & \text{if } j \in \Gamma(b, r) \\ 0 & otherwise \end{cases} \tag{3}$$

Here the neighborhood function $\Gamma(b, r)$ returns the set of neurons within the radius r centered at index b.

Within each training epoch, the steps represented by equations 1 and 2 are repeated for all $\mathbf{x}_i \in X$. The radius is initialized as $r = \sqrt{x^2 + y^2}$, that is, initially it encompasses the entire map, and shrinks until it reaches 1 (after each epoch the value is decreased by $\frac{\sqrt{x^2+y^2}}{L}$, with L the total number of iterations the algorithm is to run).

There are a number of ways of looking at the resulting map. One is as a projection of the input data onto the map. This projection allows the topography of the high-dimensional input to be preserved in the low-dimensional output space. Another interpretation is that, effectively, SOM tries to create a sample of points equivalent to the input data and we can view the training data and the trained neurons as two populations drawn from the same underlying distribution.

It is perhaps surprising for readers accustomed to the traditional Gaussian that we only consider the "bubble" neighborhood here. However, we have shown that the Gaussian leads to much longer convergence times with very little if any effects on the quality of the map [14], [15]. We also dispense with the idea of multi-phase training since it can be shown to have no effect on the quality of the map.

III. Quality Measures

The quality measures chosen for this study represent a cross section of different approaches of looking at map convergence. Some taking a more traditional data fitting approach (quantization error and population-based convergence) and others a more topology based approach (topographic error and trustworthiness/neighborhood preservation):

- The *quantization error* was first proposed by Kohonen and is computed by calculating the average distance between the nodes and the training data points [9].
- *Topographic error* accounts for a SOM's preservation of local topological features in a low dimensional output space [8].
- *Trustworthiness* and *neighborhood preservation* evaluate to what degree the neighborhoods in the projection are actually present in the input space and vice versa [17].
- *Population-based convergence* is a measure based on a statistical analysis of the map where the training data and the neurons are considered two different populations drawn from the same underlying distribution [2], [4], [6].

Detailed surveys of a variety of other methods appear in [12], [13].

A. Computational Complexity

Table I gives the computational complexity of the quality measures. Here we ignore the effect of data dimensionality and only report the complexity in terms of number of training samples n. This is a reasonable assumption given that in most cases $n \gg d$ where d is the dimensionality of the training data.

TABLE I
COMPUTATIONAL COMPLEXITY OF QUALITY MEASURES

Quality Measure	Complexity	Short Name
Qantization Error	$O(n^2)$	qe
Topographic Error	$O(n^2)$	te
Pop. Based Convergence	$O(n)$	cv
Trustworthiness	$O(n^3 log(n))$	np.trust
Neighborhood Preservation	$O(n^3 log(n))$	np.pres

IV. EXPERIMENT DESIGN

Here we provide a quantitative analysis and comparison of the quality measures defined above. It follows an empirical procedure in which the learned structure of SOMs trained with various data sets is evaluated. The experimentation procedure for each data set has two high-level steps:

1) Train a large number of maps to determine when each quality measure converges.
2) Evaluate the clustering and accuracy of converged maps and compare the results to determine how well each convergence criteria performs.

A. Training

Training a SOM using the above training algorithm requires that the number of iterations must be known in advance. Therefore, a large number of maps are trained at fixed iteration steps (i.e. powers of two) to compute the value of the quality measure at each step.

In order to determine when a quality measure has converged, we must have a concise definition of convergence. Borrowing from conventional artificial neural network training, a quality measure is converged when its rate of change between steps falls below a set threshold. More specifically, when $\Delta Q(t, d) < \epsilon$ with ΔQ defined in (4).

$$\Delta Q(t, d) = \frac{1}{d} \sum_{i=t-d}^{t} \frac{Q_i - Q_{i-1}}{Q_{i-1}} \tag{4}$$

With t the iteration step, d the number of steps to include, and Q_i the value of the quality measure at step i. The average of several steps is used to prevent a smaller than expected change between steps from causing premature convergence. For the following experiments we have $\epsilon = 0.05$ and $d = 5$. We should also mention that we set an upper limit of 2^{20} training iterations where we terminate the experiment regardless of convergence.

The overall training strategy is as follows:

- Select a map size with a number of neurons equal to approximately 75% of the training data size for each data set.
- For each iteration step a sufficiently large number of training runs is used to determine the value of the quality measure at that step. Here we construct 300 maps at each step.
- For each of the training runs at some iteration step: the data set is shuffled, the map is randomly initialized and trained, and the quality measure is calculated.
- The resulting maps and values are stored for subsequent analysis.

Note that this research is meant to compare the quality measures and not to determine the optimal parameters for SOM training. Beyond the iterations, map size, and map initialization, all other parameters (i.e. learning rate ($\alpha = 0.35$), *etc.*) are kept static.

B. Evaluation

Although the SOM algorithm uses unsupervised learning (i.e. a target attribute is not factored into the training), labeled data is used to assist in evaluating the structure of the converged maps. For the purposes of this analysis, a SOM can be interpreted in two ways: as a clustering for the input data and as a classifier for the input data. Both of these approaches are used to evaluate how well a trained map models the underlying structure of the data.

Extraction of the clustering structure of a map is accomplished by viewing the map as a planar graph in which clusters are connected components [3]. The clustering structure can then be validated against the labels of the input data by examining cluster homogeneity and completeness. Homogeneity means that only data points with the same class are assigned to the same cluster; completeness means that all data points with the same class are assigned to the same cluster. The *V-measure* [7] is an entropy-based cluster evaluation measure that reports a single score combining homogeneity and completeness.

When a map is interpreted as a classifier, the label of an input instance can be predicted by finding its best matching neuron and assigning the majority label of the data points mapped to that neuron to the input instance. We can express the quality of the map in terms of its *labeling accuracy*.

The maps converged under each quality measure will be compared using the clustering (V-measure) and classifier (labeling accuracy) views of the maps.

V. DATA

Both synthetic and real world data sets were selected to represent a variety of scenarios under which the quality measures are compared. In the following sections we briefly describe the data sets.

A. Fundamental Clustering Problem Suite

The Fundamental Clustering Problem Suite (FCPS) [16] is a collection of synthetic data sets that present various problems for clustering algorithms (e.g. overlapping clusters, linearly non-separable clusters, *etc.*) . All of the data sets have class labels and are in three dimensions which makes them ideal for both evaluating the accuracy of a clustering and visualization. The data sets we used for experimentation are *Hepta* and *Chainlink*

The *Hepta* data set was selected based on the assumption that a well-trained SOM would model the clustering well. The data set has well-defined, convex clusters and represents the simplest case. It has 212 instances and seven classes. It is visualized in Figure 2.

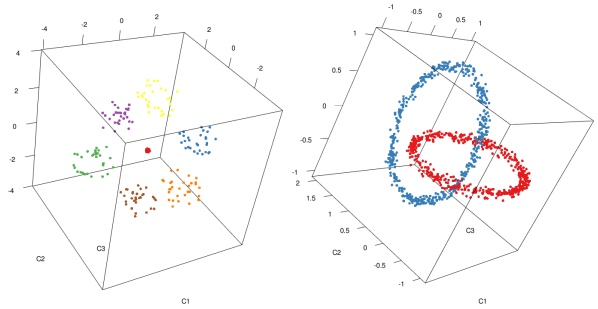

Fig. 2. *Hepta* and *Chainlink* data visualizations, respectively.

TABLE II
Ecoli CLASS COUNTS

cp	im	pp	imU	om	omL	imL	imS
143	77	52	35	20	5	2	2

The *Chainlink* data set was selected knowing that a SOM could not model it completely. The data set has interlinking rings as clusters that are non-linearly separable. It has 1000 instances and two classes. It is also visualized in Figure 2.

B. Ecoli

The *Ecoli* data set [11] is a real world data set consisting of attributes for classifying the localization site of *E. coli* proteins. The data has seven real-valued independent attributes. The dependent attribute is a categorical variable with eight levels. The data has 336 instances and the value counts for the various levels is highly unbalanced as shown in Table II.

VI. RESULTS

For each of our data sets, the convergence of the quality measures is visualized and the clustering quality and label accuracy are graphed.

A. FCPS Results

1) Hepta: Figure 3 shows the graphs with the values of each quality measure as a function of the number of iterations using a map of size 10×15 for the *Hepta* data set. The graphs display the mean and the range of each of the quality measure values. The range indicates that due to the stochastic nature of the SOM training algorithm, the outcome of the training runs with the same data set is not deterministic.

Note that all of the quality measures display a convergence behavior and can be considered to be converged after about 32,000 training iterations. Because the *Hepta* data set is the prototypical clustering data set with seven convex clusters, we can observe that the error quality measures report very small errors and the other quality measures report values close to 1 after they have converged. The convergence behavior is consistent with our convergence graph in Figure 4.

The blue line in Figure 4 represents the label accuracy (lab.acc) with respect to the number of training iterations and the red line represents the V-measure (v.measure) with respect to the number of training iterations. We then plot the

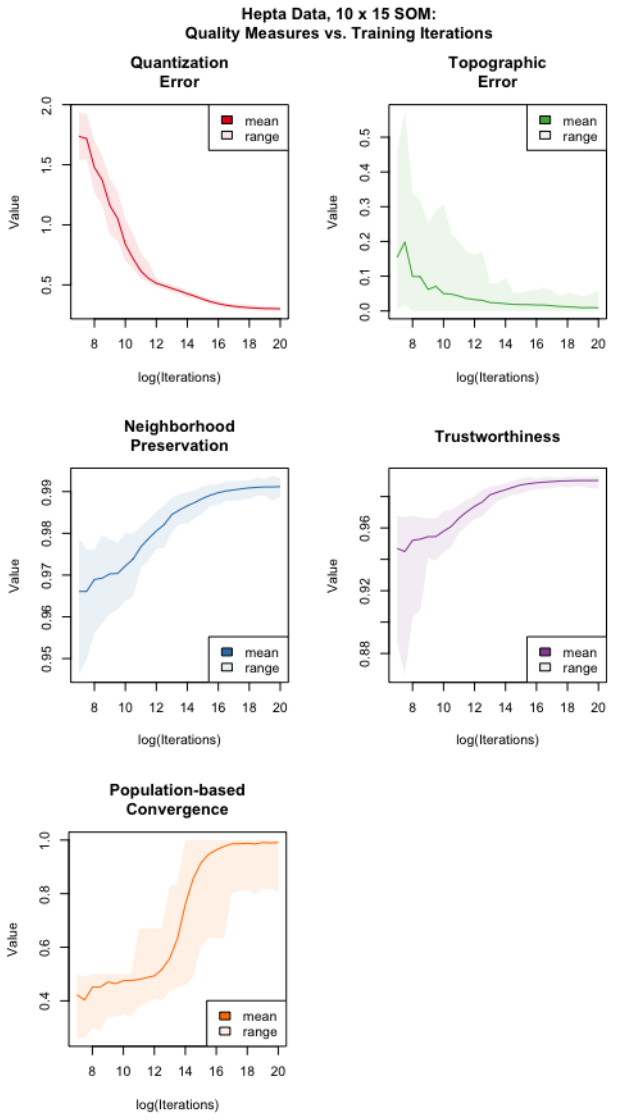

Fig. 3. *Hepta* quality measures.

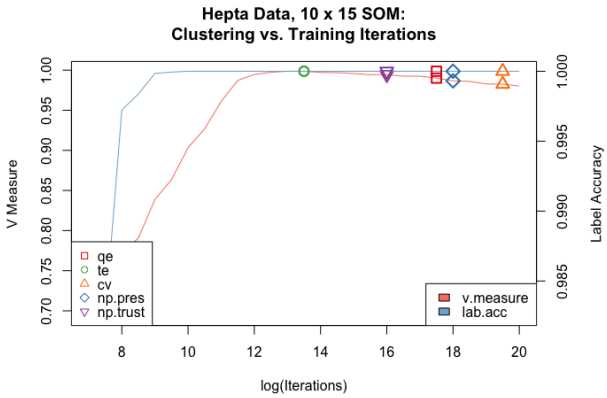

Fig. 4. *Hepta* V-measure and labeling accuracy.

convergence points of our various quality measures according to (4) against the V-measure and labeling accuracy.

What is striking is that the topographic error (te) is the most optimistic quality measure reporting convergence at about 32,000 iterations and that the population-based convergence (cv) is the most conservative quality measure reporting convergence at around 250,000 iterations. However, all of the quality measures capture the essence of SOM learning. We can easily observe that both the V-measure and the labeling accuracy are at 100% long before any of the quality measures report convergence.

The large difference in terms of reported convergence by the topographic error and the population-based convergence is due to the fact that neighborhoods and clusters start forming very early during learning but that learning the underlying data distribution takes some time. Therefore, the topographic error is a good quality estimate for the underlying cluster structure and topology but not a good estimate for the underlying data fit.

What is perhaps interesting is that the converse seems to hold: once the map has learned the underlying data distribution, it follows that it has learned the topology and neighborhoods in the data. This observation certainly holds for all the remaining experiments; and has been our observation during the running of many other experiments.

2) Chainlink: Figure 5 shows the graphs with the values of each quality measure as a function of the number of iterations using a map of size 25×30 for the *Chainlink* data set. We can easily observe a convergence behavior of the various quality measures similar to the one we observed for *Hepta*. With the exception perhaps of the graph for the population-based convergence. Here, the mean of the 300 different map does not appear to have converged at our cutoff of 1,048,576 training iterations. However, we can observe that the top end of the range envelope appears to have converged and therefore we expect that the mean would also converge with additional training iterations. Here we use the cutoff value of 2^{20} training iterations as our convergence point.

Recall that the *Chainlink* data set is made up of two interlocking rings that are not linearly separable. Therefore, the structure cannot be completely modeled by the SOM. Observe that only the population-based convergence reports this fact by converging on a value much less than 1 (we can assume that because the upper bound of the range envelope has converged to the value 0.7); implying that the SOM cannot represent the underlying data distribution appropriately. All other quality measures converge on an almost perfect score, which seems to highlight the fact that assessing the quality of a map by only looking at local structures leads to overly optimistic assessments.

The V-measure and labeling accuracy are shown in Figure 6. Note that the V-measure does not exceed 0.45, again highlighting the fact that the interlocking rings cannot be represented appropriately by the SOM. This is consistent with the finding using the population-based convergence.

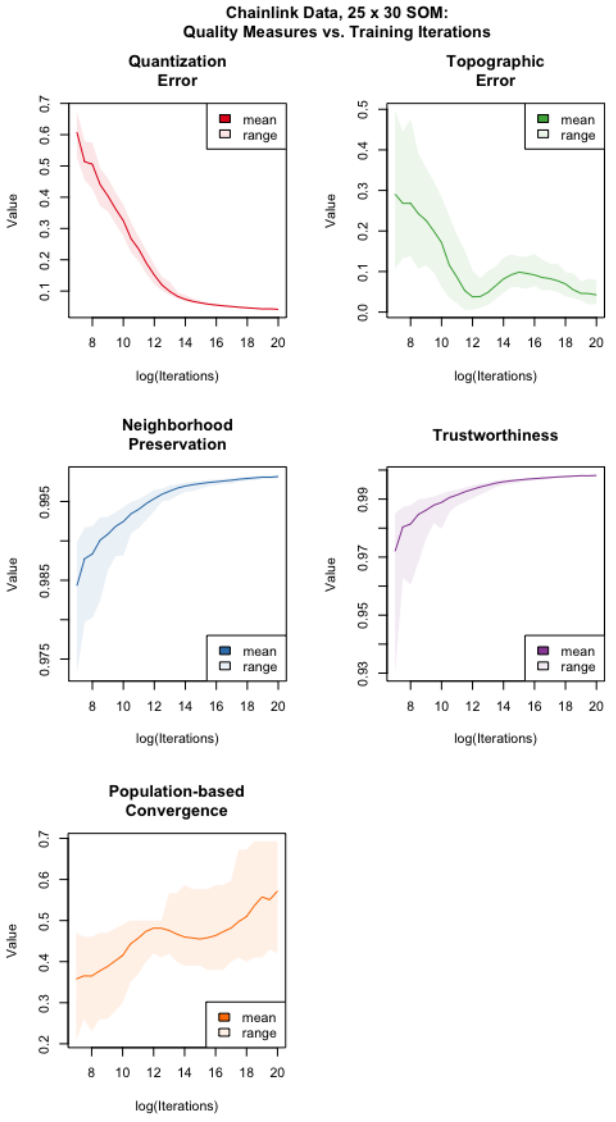

Fig. 5. *Chainlink* quality measures.

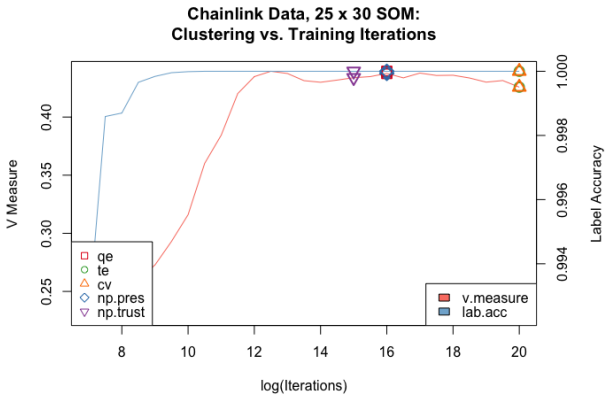

Fig. 6. *Chainlink* V-measure and labeling accuracy.

Surprisingly, the labeling accuracy for this map is at 100% after only about 1,000 iterations. The discrepancy between the V-measure and the labeling accuracy can be explained by the fact that labels for an input are computed by looking at the majority label at each neuron.

From Figure 6 we can see that the population-based convergence quality measure (cv) and the topographic error (te) are the most conservative ones. Our observation that convergence of the population-based quality measure implies that all other quality measures have also converged still holds.

B. *Ecoli* Results

Figure 7 shows the graphs with the values of each quality measure as a function of the number of iterations using a map of size 14×18 for the *Ecoli* data set. We can observe the now familiar convergence pattern for each of the quality measures, again with the exception perhaps of the graph for the population-based convergence. As before we can observe that the top end of the range envelope appears to have converged and therefore we expect that the mean would also converge with additional training iterations. Here we use the cutoff value of 2^{20} training iterations as our convergence point.

All of the quality measures with the exception of the population-based convergence report near perfect scores after convergence. This is somewhat disconcerting because a peek at Figure 8 shows that neither the V-measure nor the labeling accuracy reach 100%. Only the mean of the population-based convergence reports a value of about 0.9 of the top of the range envelope at the cut-off indicating that the map cannot model the given cluster structure completely.

The *Ecoli* data set comprises eight clusters in 7-dimensional space where some of the clusters are very small (see Table II). It is therefore not surprising that the map cannot completely model this cluster structure. Figure 1 shows a slightly larger map at 15×20 of the *Ecoli* data set after 100,000 iterations with a population-based convergence of 0.9. It is easy to see that of the eight existing clusters the map only modeled three (light areas surrounded by darker areas). With a close inspection of the labels that appear in each of the clusters, we can see that only the three larger classes were modeled and that the clusters themselves are not homogeneous. This reinforces the findings of Figure 8 that a complete modeling of this data set is not possible with this size map.

Figure 8 also confirms what we have seen before, that the population-based convergence is the most conservative of all the quality measures. Meaning that convergence of this measure implies convergence of all the other measures.

C. *Observations and Remarks*

The previous set of experiments has shown that the population-based convergence is the most conservative of the quality measures studied here in a twofold sense:

1) The convergence of this quality measure implies convergence of all the other measures studied here, and

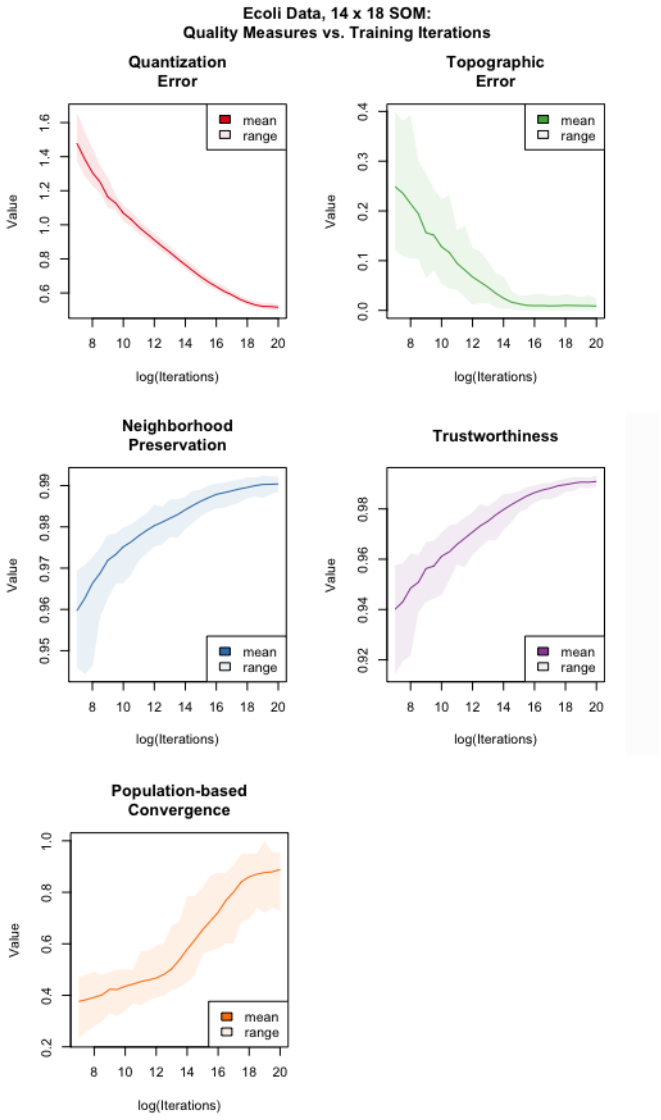

Fig. 7. *Ecoli* quality measures.

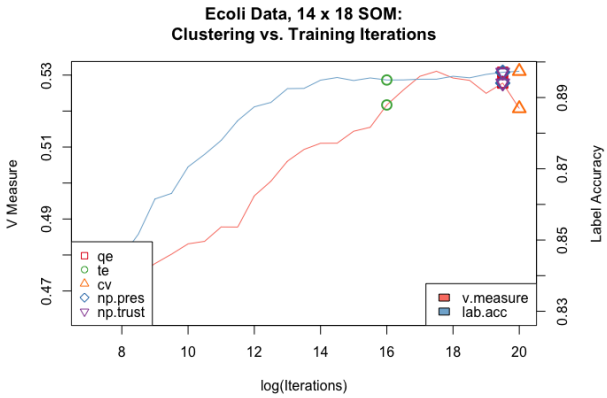

Fig. 8. *Ecoli* V-measure and labeling accuracy.

2) it is conservative in its estimate of the quality of the produced map and is consistent with the V-measure analysis of the learned cluster structures.

The latter point is particularly important because ultimately we would like to use these measures to tell us something about the quality of the maps that we are constructing. If we are using a measure that is overly optimistic, we are tempted to make unwarranted inferences on the cluster structures displayed on the induced map.

Here we have kept a number of parameters static. One of question one might ask is: is it not possible that some of these parameters may affect the evaluated measures differently? This is certainly possible. However, it does not matter if on other maps the measures would behave differently. We have shown that they do not work properly in this setting and therefore are unreliable according to Popper's theory falsification [21].

Our findings here reinforce our earlier findings [19] that population-based convergence is more conservative than neighborhood stability [20].

From Table I we see that population-based convergence is computationally the most efficient quality measure where execution time grows linearly with the number of training samples. Compare this to the exponential nature of trustworthiness and neighborhood preservation which makes these measures practical only for the smallest of training sets. The complexity of quantization and topographic error grows as the square of the number of training examples, again limiting their usefulness for large data sets. In [2] we proposed the *estimated topographic accuracy* as a computationally efficient analog to the topographic error.

VII. Conclusions

By looking at our five quality measure values as a function of training iterations and evaluating the values they report against cluster quality assessments such as the V-measure and labeling accuracy, we found that, with the exception of the population-based convergence, the quality measures were too optimistic in the sense that they reported near perfect scores for maps that were demonstrably far from perfect. This has far reaching consequences in that the user might be tempted to make unwarranted inferences on the cluster structures displayed on the induced map reported to be perfectly converged if, in fact, it has not.

Another result that surprised us is that reporting convergence on the underlying distribution using the population-based convergence implies convergence of all the other quality measures. In hindsight this should perhaps be obvious in the sense that, in an attempt to model the underlying distribution, the SOM will in fact reconstruct clusters and neighborhoods – the targets of many of the proposed quality measures.

Finally, it turns out that the population-based convergence is the most computationally efficient quality measure considered here. Its execution time grows linearly with the size of the training data, making it an extremely practical tool for the evaluation of maps.

We have implemented the population-based convergence in our *popsom* R-package [6]. We will be releasing a Python-based version of this package in the near future.

References

[1] Bauer, H.-U. and Pawelzik, K., "Quantifying the neighborhood preservation of self-organizing feature maps," *IEEE Trans. Neural Netw.*, vol. 3, no. 4, pp. 570–579, 7 1992.

[2] Hamel, L., "Som quality measures: an efficient statistical approach," in *Advances in Self-Organizing Maps and Learning Vector Quantization, Proc. 11th International Workshop WSOM 2016*, Houston, TX, 2016, pp. 49–59.

[3] Hamel, L. and Brown, C., "Improved interpretability of the unified distance matrix with connected components," in *Proc. International Conf. Data Mining*, Las Vegas, NV, 2011, pp. 338–343.

[4] Hamel, L. and Ott, B., "Population-based convergence criterion for self-organizing maps," in *Proc. International Conf. Data Mining*, Las Vegas, NV, 2012, pp. 98–104.

[5] Hamel, L., "Som training: a modern view," 2015, unpublished.

[6] Hamel, L., Ott, B., Breard, G., Tatoian, R., and Vishakh, G. (2017). "popsom: Functions for Constructing and Evaluating Self-Organizing Maps," *R package version 4.2*. https://CRAN.R-project.org/package=popsom

[7] Hirschberg, J. and Rosenberg, A., "V-measure: A conditional entropy-based external cluster evaluation measure," in *EMNLP-CoNLL*, vol. 7, 2007, pp. 410–420.

[8] Kiviluoto, K., "Topology preservation in self-organizing maps," in *Proc. International Conf. Neural Networks*, Washington, DC, 1996, pp. 294–299.

[9] Kohonen, T., *Self-organizing maps*. Berlin, Germany: Springer, 2001.

[10] Lampinen, J. and Oja, E., "Clustering properties of hierarchical self-organizing maps," in *Mathematical Nonlinear Image Processing*. Springer, 1993, pp. 165–176.

[11] Lichman, M., "Ecoli," in *UCI Machine Learning Repository*. School Inform. and Comput. Sci., Univ. California, Irvine, 2013, [Dataset].

[12] Polani, D., "Measures for the organization of self-organizing maps," in *Self-Organizing Neural Networks*, 1st ed., Seiffert, U. and Jain, L., Eds. Heidelberg, Germany: Physica-Verlag, 2002, pp. 13–44.

[13] Pölzlbauer, G., "Survey and comparison of quality measures for self-organizing maps," in *Proc. 5th Workshop Data Analysis*, Slovakia, 2004, p. 6782.

[14] Tatoian, R. and Hamel, L., "Self-organizing map convergence," in *Proc. International Conf. Data Mining*, 2016, p. 92.

[15] Tatoian, R. and Hamel, L., "Self-Organizing Map Convergence," in *the International Journal of Service Science, Management, Engineering, and Technology (IJSSMET)*, IGI Global, Vol. 9, Issue 2, pp 61-85, 2018.

[16] Ultsch, A., "Clustering with som: U*c," in *Proc. Workshop on Self-Organizing Maps*, Paris, France, 2012, pp. 75–82, [Dataset].

[17] Venna, J. and Kaski, S., "Neighborhood preservation in nonlinear projection methods: An experimental study," *Lecture Notes in Comput. Sci.*, vol. 2130, p. 485491, 2001.

[18] Villmann, T., Der, R., Herrmann, M., and Martinetz, T., "Topology preservation in self-organizing feature maps: exact definition and measurement," *IEEE Trans. Neural Netw.*, vol. 8, no. 2, pp. 256–266, 3 1997.

[19] Ott, B., "A Convergence Criterion for Self-Organizing Maps," MS Thesis, University of Rhode Island, 2012.

[20] Cottrell, M., De Bodt, E., and Verleysen, M., "A statistical tool to assess the reliability of self-organizing maps," Advances in self-organising maps, p. 714, Springer, 2001.

[21] Popper, K., "The logic of scientific discovery," Routledge, 2005.