Incipient Faults Diagnosis by Employing Self Organizing Map

Dr. Tarun Chopra

Associate Professor, Department of Electrical Engineering
Govt. Engineering College Bikaner, - 334004, INDIA
E-mail: *tarun_ecb@rediffmail.com*

Abstract: The Self Organizing Map (SOM) plays a versatile role in providing an initial organization of the data and is an epistemological tool for acquiring an understanding of the semantics of data and for generating hypothesis about the associated faults. An important aspect of the SOM based architecture is that use of SOM as constructive learning based epistemic tool has encouraged the researcher to take up of more complex issues.

Keywords: DAMADICS Benchmark Process Control System, Fault Diagnosis, SOM

I. INTRODUCTION

Statistical, Machine Learning and Neural Network based techniques have been found to be fairly effective for abrupt faults. However, it has been found that the application of these techniques to the datasets representing the entire spectrum of possible faults produces limited success. Apparently, the classification of incipient faults has turned out to be a fairly complex decision making problem due to nonlinear relationships among the high dimensional data related to different measured parameters. Hence, further research is required to be focused on developing a suitable methodology based on an epistemological tool for acquiring an understanding of the semantics of data and for generating hypothesis about the associated faults.

SOM based method has been found to be useful for mapping the data onto its fault class. In order to make this information explicit, it is proposed to utilize the abilities of SOM for visualization of high-dimensional data [1].

Hence, for dealing the complex problem of separation of incipient faults of highly overlapping nature, an epistemological decision making approach using SOM is proposed in this paper.

II. PROPOSED METHODOLOGY

An innovative methodology employing SOM based fault diagnosis is proposed in this research for the incipient fault data received from the output stage of hybrid classifier in Secondary Decision Making System.

In the first stage of proposed methodology, the original raw data is prepared in the preprocessing phase and transformed by normalization.

A brief description of the basic working principle of SOM based technique and algorithm for the proposed scheme of Computational Decision Making for Incipient Faults has been presented here.

The items in the input data set are assumed to be in a vector format. If n is the dimension of the input space, then every node on the map grid holds an n-dimensional vector of weights.

$$m_i = [\, m_{i1} , m_{i2} , m_{i3} , \dots , m_{in}]$$

The basic principle of the SOM [2] is to adjust these weight vectors until the map represents "a picture" of the input data set. Since the number of map nodes is significantly smaller than the number of items in the dataset, it is impossible to represent every input item from the data space on the map. Rather, the objective is to achieve a configuration in which the distribution of the data is reflected and the most important metric relationships are preserved.

The SOM training algorithm involves essentially two processes, namely

- Vector quantization
- Vector projection

Vector quantization creates a representative set of output vectors from the input vectors. In general, vector quantization reduces the number of vectors. This can be considered as a classification, or clustering, process.

Vector projection aims at projecting output vectors (in d-dimensional space) onto a regular tessellation in lower dimensions (i.e., a SOM), where the regular tessellation consists of an arbitrary number of neurons. Each output vector is projected onto a neuron such that the "close" output vectors in d-dimensional space will be projected onto neighboring neurons in the SOM. This will ensure that the initial pattern of the input data will be preserved in the map formed by the neurons.

These two tasks are illustrated in figure 1.

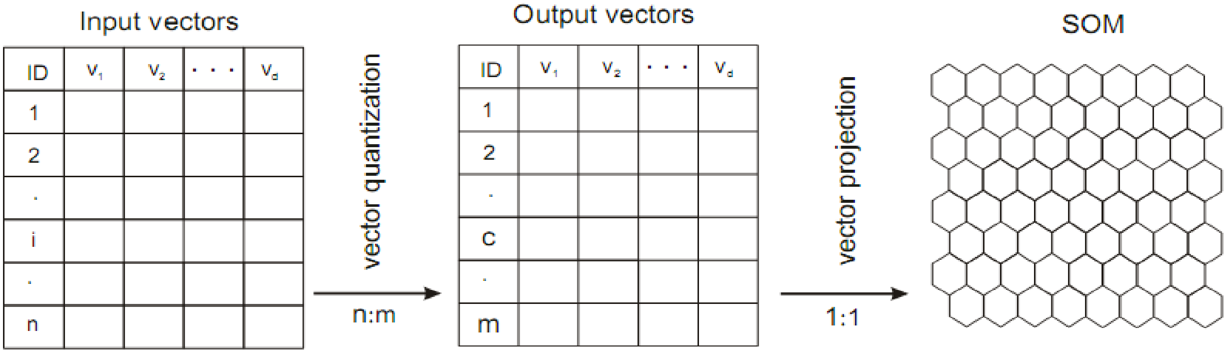


Figure 1: Illustration of SOM Principles

III. Application of Proposed Methodology to DAMADICS Problem

The efficacy of proposed methodology has been demonstrated for the group { F3,F4,F5,F6,F9} of incipient fault sets of DAMADICS Problem considering all measured parameters:-

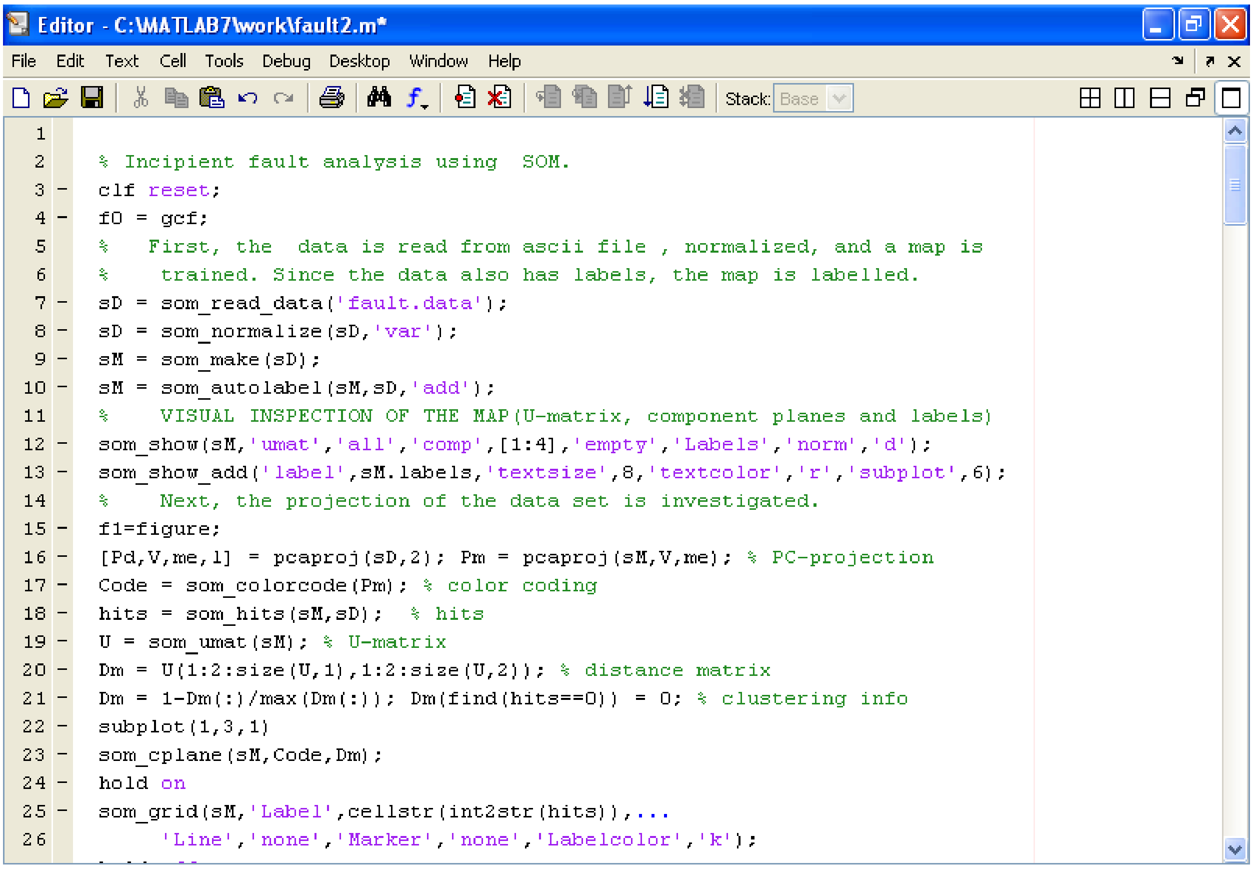
Application of Proposed Methodology to Group of Incipient Faults:

The performance of SOM based fault classifier has been tested for demonstrating its visualization abilities for the sample fault data set pertaining to the fault classes F3,F4,F5,F6 and F9 briefly discussed in table1.

Table 1: Overlapping Fault category {F3, F4, F5, F6, F9}

S. No	***Fault***	***Location***	***Description***
1	F3	Control Valve	Valve plug or valve seat erosion
2	F4	Control Valve	Increased of valve or bushing friction
3	F5	Control Valve	External leakage
4	F6	Control Valve	Internal leakage
5	F9	Servo-motor	Servo-motor's housing or terminals tightness

The analysis has been made using SOM Toolbox [4] in MATLAB environment. The screen shot of program is shown in figure 2.


Editor - C:\MATLAB7\work\fault2.m*
File Edit Text Cell Tools Debug Desktop Window Help
Stack: Base

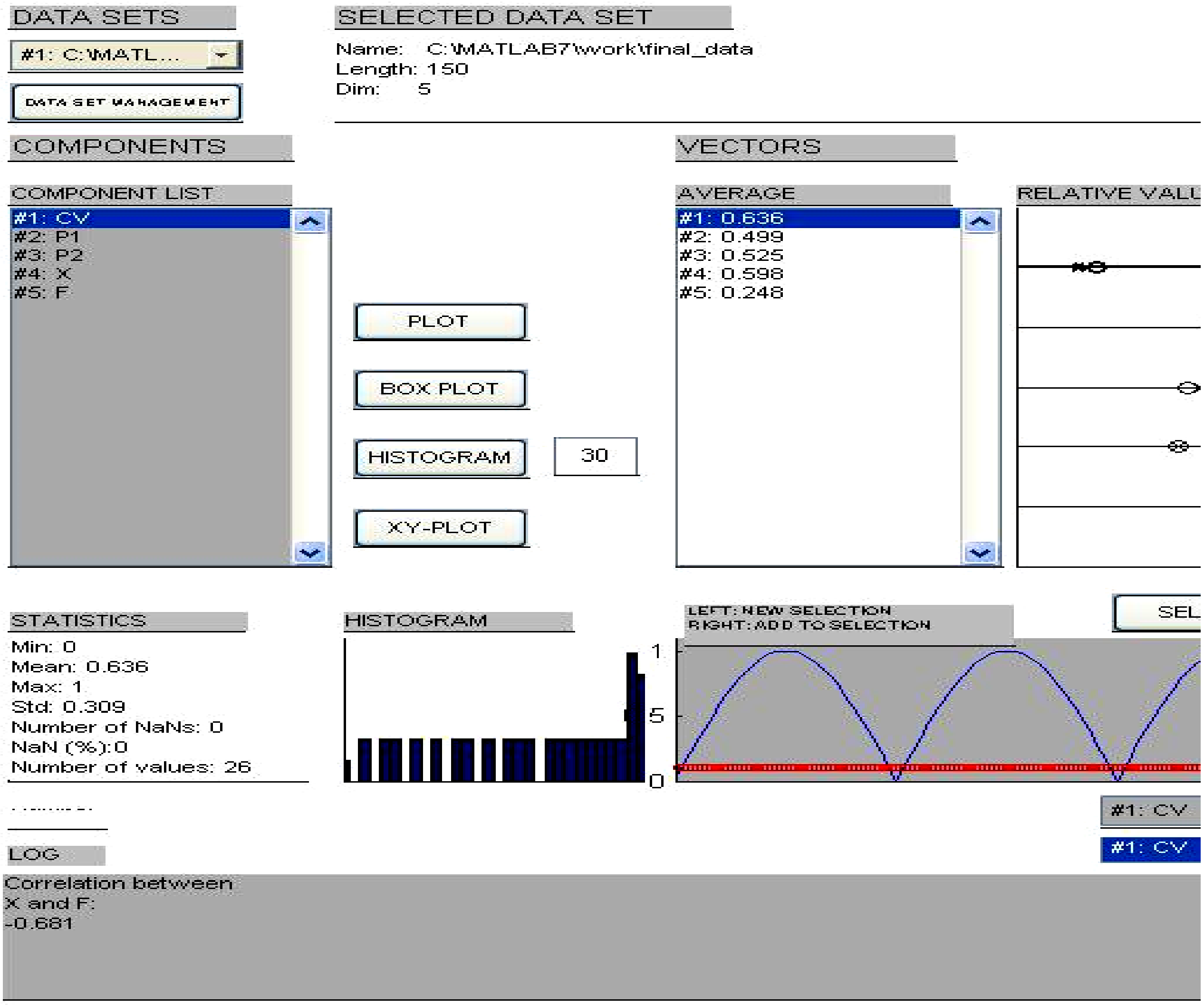
```
% Incipient fault analysis using  SOM.
clf reset;
f0 = gcf;
%   First, the  data is read from ascii file , normalized, and a map is
%    trained. Since the data also has labels, the map is labelled.
sD = som_read_data('fault.data');
sD = som_normalize(sD,'var');
sM = som_make(sD);
sM = som_autolabel(sM,sD,'add');
%    VISUAL INSPECTION OF THE MAP(U-matrix, component planes and labels)
som_show(sM,'umat','all','comp',[1:4],'empty','Labels','norm','d');
som_show_add('label',sM.labels,'textsize',8,'textcolor','r','subplot',6);
%    Next, the projection of the data set is investigated.
f1=figure;
[Pd,V,me,l] = pcaproj(sD,2); Pm = pcaproj(sM,V,me); % PC-projection
Code = som_colorcode(Pm); % color coding
hits = som_hits(sM,sD);  % hits
U = som_umat(sM); % U-matrix
Dm = U(1:2:size(U,1),1:2:size(U,2)); % distance matrix
Dm = 1-Dm(:)/max(Dm(:)); Dm(find(hits==0)) = 0; % clustering info
subplot(1,3,1)
som_cplane(sM,Code,Dm);
hold on
som_grid(sM,'Label',cellstr(int2str(hits)),...
     'Line','none','Marker','none','Labelcolor','k');
```

Figure 2: Screen shot of MATLAB Program

1) Preprocessing of Dataset

The data set consisting of 50 representative samples from each of five types of faults (a total of 250 samples), is read from ASCII file. The measured variables are CV (process control external signal), P1 (pressure on valve inlet), P2 (pressure on valve outlet), T (Temperature) X (valve plug displacement), F (main pipeline flow rate). The label associated with each sample is the fault type information viz ‘F3’ (Valve plug or valve seat erosion), ‘F4’ (Increased of valve or bushing friction) etc.

The different components of the data set are usually normalized in the Pre-processing of Dataset, as shown in figure 3.



Name: C:\MATLAB7\work\final_data
Length: 150
Dim: 5

#1: CV
#2: P1
#3: P2
#4: X
#5: F

#1: 0.636
#2: 0.499
#3: 0.525
#4: 0.598
#5: 0.248

Min: 0
Mean: 0.636
Max: 1
Std: 0.309
Number of NaNs: 0
NaN (%):0
Number of values: 26

Correlation between
X and F:
-0.681

Figure 3: Preprocessing of Data Set

The results of preprocessing and normalization of dataset have been shown in tables 2.

Table 2: Preprocessing information

Measured Parameter	**Mini-mum**	**Maxi-mum**	**Mean**	**Std. Deviation**
CV	0.5157	0.7378	0.6434	0.0697
P1	0.8323	0.9187	0.875	0.031
P2	0.6428	0.6573	0.6498	0.005
F	0.2118	0.2171	0.2147	0.0012
T	0.5005	0.7434	0.6464	0.0774
X	0.1995	0.7883	0.3867	0.1748

2) Map Training

The function SOM_MAKE is used to train the SOM [4]. It first determines the map size, then initializes the map using linear initialization, and finally uses batch algorithm to train the map in following steps:-

- Determination of map size
- Initialization
- Training using batch algorithm
- Rough training phase
- Fine tuning phase

The qualitative and quantitative analysis of results obtained by using the proposed methodology is presented in following Subsections**.**

3) Results

The results for the chosen data set after Map training step are obtained as follows:-

- Map size = [10, 5] i.e., A two-dimensional SOM of 50 neurons (10 by 5), organized in a hexagonal neighborhood lattice.

After the fine tuning phase, the following results are obtained for the data set:

- ○ Quantization error: 0.683
- ○ Topographic error: 0.000

4) Map Analysis by Visual Inspection

The first step in the analysis of the map is visual inspection. The U-matrix and component planes obtained for the dataset are shown in figure 4.

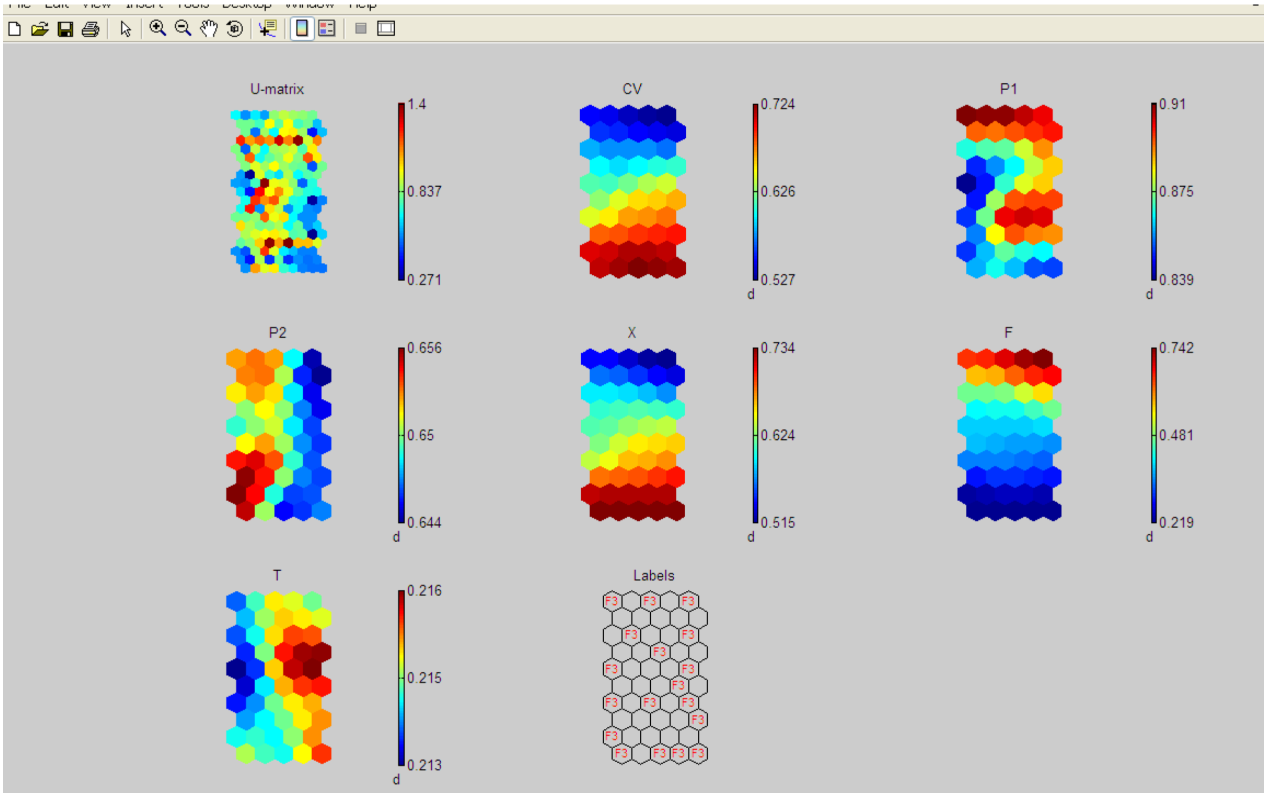


Figure 4: Visualization of U-matrix and Component Planes

The Unified distance matrix (U-matrix) is useful for detection of cluster borders and especially suitable for estimation of inter cluster distances. The component planes ('CV', 'P1', 'P2', ‘X’ 'F' and ‘T’) show values possessed by the prototype vectors of the map units. The value is indicated with color, and the color bar on the right shows what the colors mean. The component plane is used to find pairs and groups of related variables. The technique is very useful when dealing with a large number of variables. The SOM does not utilize class information during the training phase. Figure 5 clearly identifies the fault labels associated with each map unit (F3, F4, F5, F6 and F9). From the labels it can be seen that unlabeled units indicate cluster borders and the map unit in the Upper half corresponds to the F3 and F4. The remaining three fault conditions form the other clusters. The U-matrix shows no clear separation between them, but from the labels it seems that they correspond to two sub clusters.

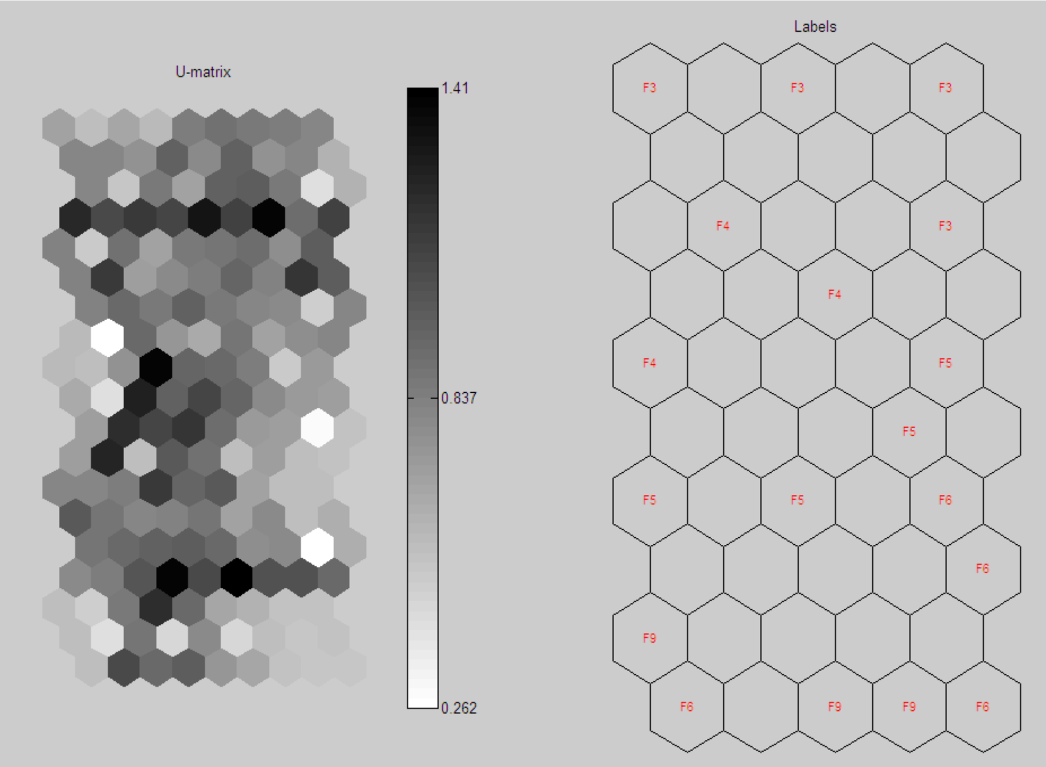


Figure 5: Visualization of U –Matrix and Class Labels

The histograms and scatter plots of the six variables used in data set are shown in figure 6. This visualization depicts quite a lot of information regarding distributions of single and pairs of variables both in the data and in the map.

From above table, it is clear that components CV and X are highly correlated.

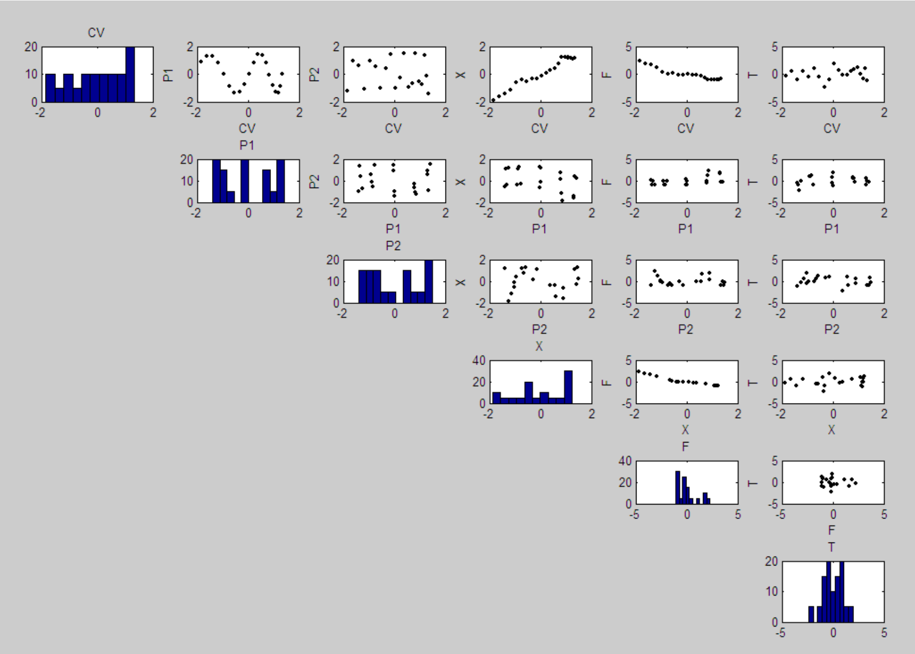


Figure 6: Histograms and Scatter Plots

5) Visualization of Projections

Next, the projection of the data set has been investigated. A principal component projection is made for the data, and applied to the map. Distance matrix information is extracted from the U-matrix [5], and it is modified by knowledge of zero-hits (interpolative) units.

Finally, in Figure 7 three visualizations are shown: the color code (with clustering information and the number of hits in each unit), the projection and the labels. Neighbouring map units are joined with lines to show the SOM topology.

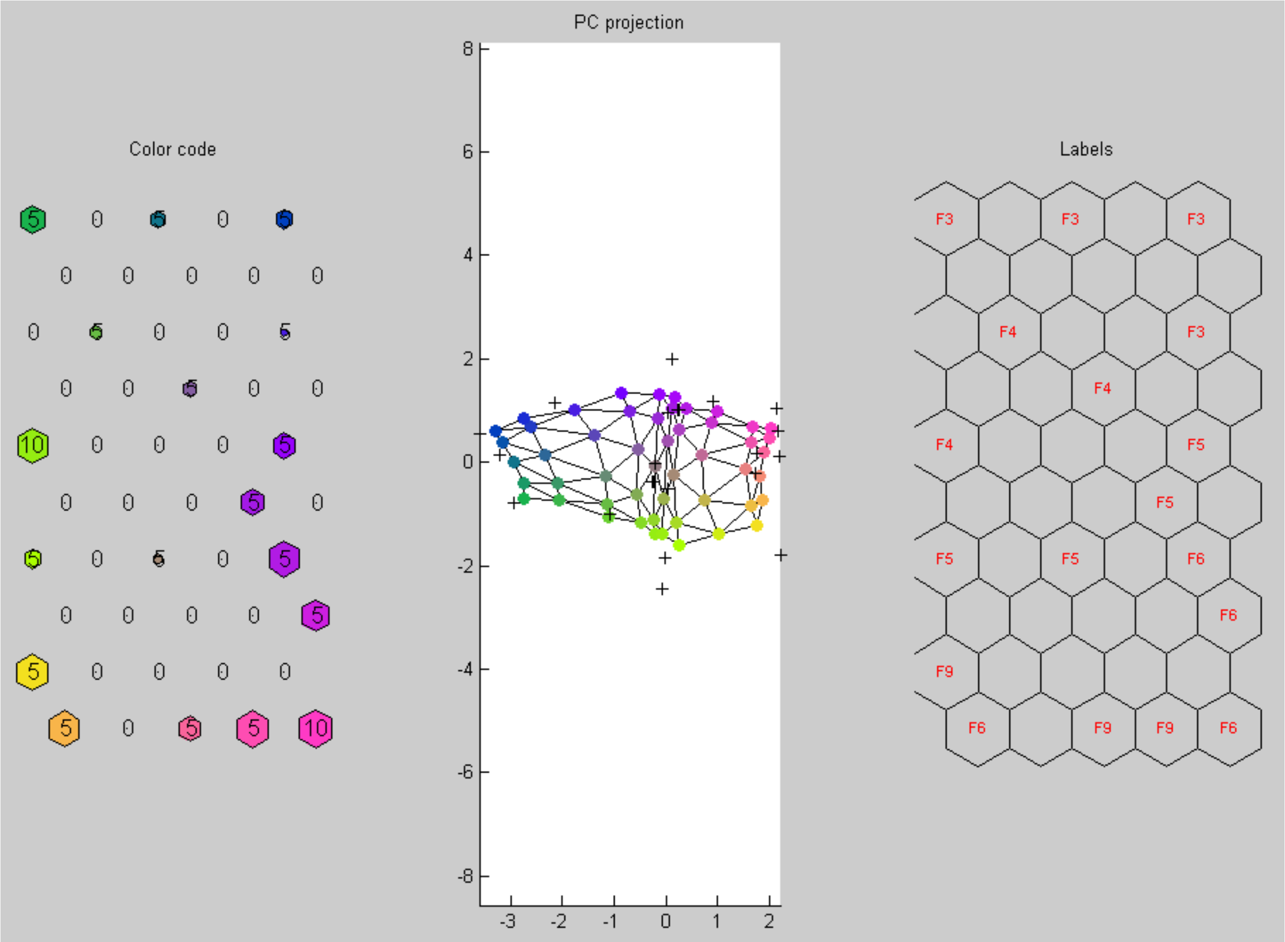


Figure 7: Visualization of Projections

6) Clustering of the Map

Visual inspection has already depicted that there are clusters in the data and the properties of these clusters are different from each other. For further investigation, the map needs to be partitioned i.e. to facilitate analysis of the map and the data; similar units need to be grouped to reduce the number of clusters [5]. This is due to the topological ordering of the unit maps. Here, the KMEANS_CLUSTERS function has been used to find an initial partitioning. Figure 8 shows the Davies-Bouldin clustering index, which is minimized with best clustering.

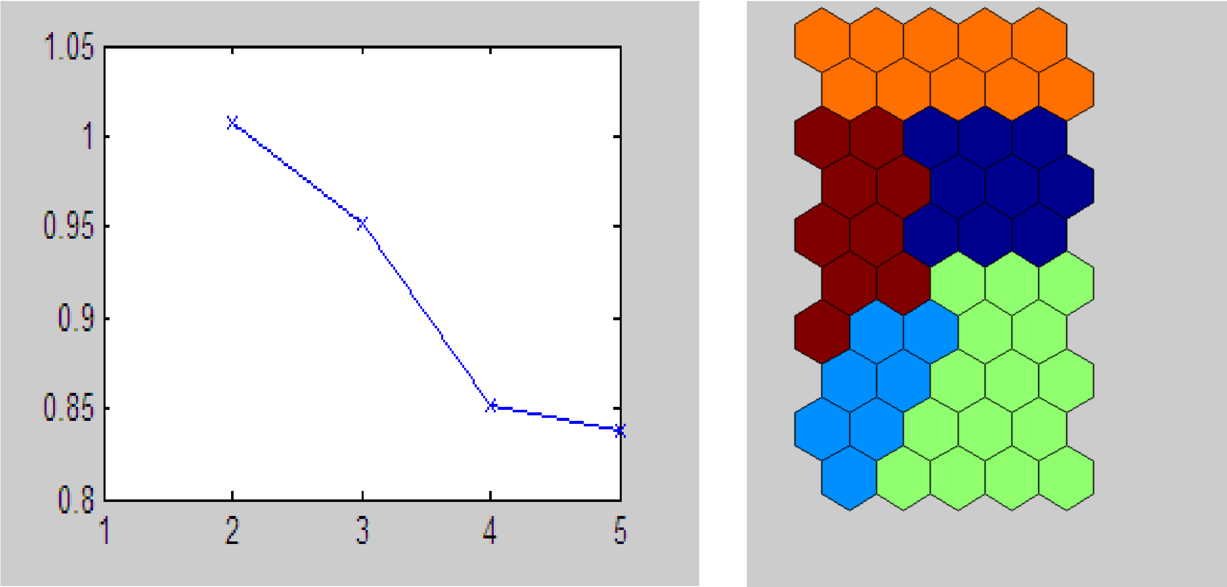


Figure 8: Results of Clustering

The Davies–Bouldin index (DB) is a metric for evaluating clustering algorithms and given by:

$$DB = \frac{1}{n}\sum_{i=1}^{n}\max_{i\neq j}\left\{\frac{S_n(Q_i)+S_n(Q_j)}{S(Q_i,Q_j)}\right\}$$

Where, n - Number of clusters, S_n - average distance of all objects from the cluster to their cluster centre, $S(Q_i Q_j)$ - distance between clusters centres. Hence the ratio is small if the clusters are compact and far from each other. Consequently, Davies-Bouldin index will have a small value for a good clustering.

The Davies-Bouldin index (0.825) obtained in this case seems to indicate that there are five clusters on the map corresponding to faults F3, F4, F5, F6 and F9 with some percentage of overlapping.

7) *Classification*

Although the SOM can be used for classification, it is important to note that it does not utilize class information at all, thus, making its results inherently suboptimal. However, using function SOM_SUPERVISED, the network can take the class information into account.

Consequent upon classification, U- Matrix obtained indicates clear cut separation between five categories of faults F3, F4, F5, F6 and F9 as shown in Figure 9.

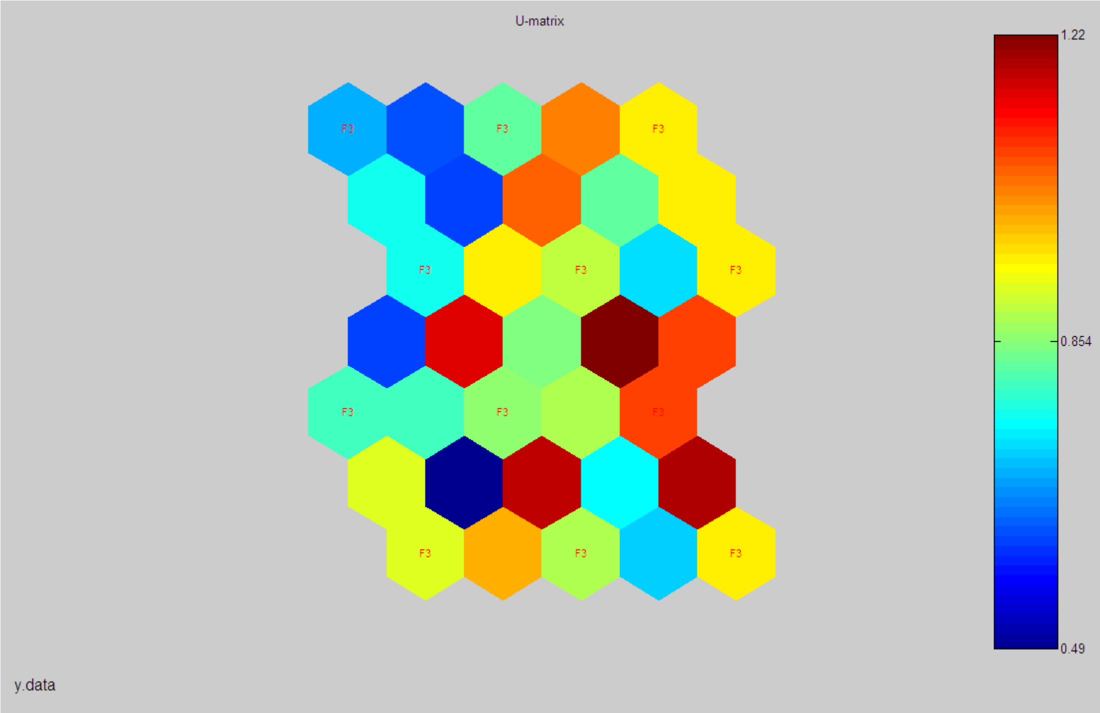


Figure 9: SOM after Supervised Learning

After performing the classification task, classification accuracy of 80.8% is obtained for the incipient fault data set.

IV. INFERENCE

The nature of data pertaining to incipient faults is generally overlapping with normal operating condition on one hand and abrupt fault condition on the other hand. Hence, it is essential to first understand the data that is being processed for obtaining better and meaningful results. The central task for gaining the necessary understanding is data exploration, which has been attempted by using SOM.

V. REFERENCES

[1] C. Aggarwal, “On k-Anonymity and the Curse of Dimensionality, ”Proc. Int’l Conf. Very Large Data Bases (VLDB), pp. 901-909, 2005.

[2] B.-C. Chen, K. LeFevre, and R. Ramakrishna, “Privacy Skyline: Privacy with Multidimensional Adversarial Knowledge,” Proc.Int’l Conf. Very Large Data Bases (VLDB), pp. 770-781, 2007.

[3] I. Dinur and K. Nissim, “Revealing Information while Preserving Privacy,” Proc. ACM Symp. Principles of Database Systems (PODS), pp. 202-210, 2003.

[4] C. Dwork, “Differential Privacy,” Proc. Int’l Colloquium Automata, Languages and Programming (ICALP), pp. 1-12, 2006.

[5] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating Noise to Sensitivity in Private Data Analysis,” Proc. Theory of Cryptography Conf. (TCC), pp. 265-284, 2006.

[6] B.C.M. Fung, K. Wang, and P.S. Yu, “Top-Down Specialization for Information and Privacy Preservation,” Proc. Int’l Conf. Data Eng. (ICDE), pp. 205-216, 2005.

[7] G. Ghinita, Y. Tao, and P. Kalnis, “On the Anonymization of Sparse High-Dimensional Data,” Proc. IEEE 24th Int’l Conf. Data Eng. (ICDE), pp. 715-724, 2008.

[8] Y. He and J. Naughton, “Anonymization of Set-Valued Data via Top-Down, Local Generalization,” Proc. Int’l Conf. Very Large Data Bases (VLDB), pp. 934-945, 2009.

[9] D. Kifer and J. Gehrke, “Injecting Utility into Anonymized Data Sets,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD), pp. 217-228, 2006.

[10] N. Koudas, D. Srivastava, T. Yu, and Q. Zhang, “Aggregate Query Answering on Anonymized Tables,” Proc. IEEE 23rd Int’l Conf. Data Eng. (ICDE), pp. 116-125, 2007.

[11] K. LeFevre, D. DeWitt, and R. Ramakrishna, “Incognito: Efficient Full-Domain k-Anonymity,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD), pp. 49-60, 2005.

[12] K. LeFevre, D. DeWitt, and R. Ramakrishna, “Mondrian Multidimensional k-Anonymity,” Proc. Int’l Conf. Data Eng. (ICDE), p. 25, 2006.

[13] K. LeFevre, D. DeWitt, and R. Ramakrishna, “Workload-Aware Anonymization,” Proc. ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), pp. 277-286, 2006.

[14] N. Li, T. Li, and S. Venkatasubramanian, “t-Closeness: Privacy Beyond k-Anonymity and ‘-Diversity,” Proc. IEEE 23rd Int’l Conf. Data Eng. (ICDE), pp. 106-115, 2007.

[15] T. Li and N. Li, “Injector: Mining Background Knowledge for Data Anonymization,” Proc. IEEE 24th Int’l Conf. Data Eng. (ICDE), pp. 446-455, 2008.

[16] T. Li and N. Li, “On the Tradeoff between Privacy and Utility in Data Publishing,” Proc. ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), pp. 517-526, 2009.

[17] T. Li, N. Li, and J. Zhang, “Modeling and Integrating Background Knowledge in Data Anonymization,” Proc. IEEE 25th Int’l Conf. Data Eng. (ICDE), pp. 6-17, 2009.

[18] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, “‘l-Diversity: Privacy Beyond k-Anonymity,” Proc. Int’l Conf.Data Eng. (ICDE), p. 24, 2006.

[19] D.J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J.Y.Halpern, “Worst-Case Background Knowledge for Privacy-Preserving Data Publishing,” Proc. IEEE 23rd Int’l Conf. Data Eng. (ICDE), pp. 126-135, 2007.