Transactions on Machine Learning
and Data Mining
Vol. 3, No. 2 (2010) 51-66
© ISSN: 1865-6781 (Journal),
ISBN: 978-3-940501-19-6,
IBaI Publishing ISSN 1864-9734

ibai Publishing
www.ibai-publishing.org

Creating Product Maps with Self-Organizing Maps for Purchase Decision Making

Kazuhiro Kohara and Tetsuya Tsuda

Department of Electrical, Electronics and Computer Engineering,
Chiba Institute of Technology
2-17-1 Tsudanuma, Narashino, Chiba 275-0016, Japan
E-mail: kohara.kazuhiro@it-chiba.ac.jp

Abstract. We propose a way of creating product maps with self-organizing maps (SOMs) for purchase decision making. We previously proposed a way of purchase decision support using SOMs and the Analytic Hierarchy Process (AHP). We provided several class boundaries, which divided the input features into several classes before creating self-organizing product maps. Because the number of classes and their boundaries depended on the person classifying the classes, the product maps were not always the same. In this paper, we first provide two class boundaries, which divide the range between the maximum and minimum of an input feature value into three equal parts. Second, we create self-organizing product maps using the classified data inputs. We applied our way to five kinds of products and confirmed its effectiveness.

Keywords: product maps, self-organizing maps, Analytic Hierarchy Process, clustering, visualization, purchase decision making, consumer buying process

1 Introduction

As reported by [Kotler 2002], marketing researchers have developed a *stages model* of the buying decision process. The consumer passes through five stages: problem recognition, information search, evaluation of alternatives, purchase decision, and postpurchase behavior. Five successive sets are involved in the consumer decision making. The first set is the *total set* of brands available to the consumer. The individual consumer knows only a subset of these brands (*awareness set*). Some brands meet the initial buying criteria (*consideration set*). As the person gathers more information, only a few brands will remain as strong contenders (*choice set*). The brands in the choice set might all be acceptable. The person makes a final choice from this set. Several intelligent decision support systems (DSSs) have been proposed to solve the variety of problems related to making decisions (e.g., [Park 2002], [Riordan 2002], [Kohara 2002], [Ha 2003], [Walle 2003], [Suka 2003], [Kohara 2006]).

We previously proposed a way of purchase decision support [Kohara 2006] using self-organizing maps (SOMs) [Kohonen 1995] and the Analytic Hierarchy Process (AHP) [Saaty1980]. First, we divided many products (*total set*) into several clusters using SOM. Second, we selected some alternatives (*choice set*) using the product maps. Finally, we made a final choice from the alternatives using AHP. As an example of real-world applications, we applied our way to the problem of buying a personal computer (PC). We considered 120 kinds of notebook PCs sold in Japan in June 2004. We clustered these PCs using the following features: CPU speed (GHz), main memory capacity (MB), HDD storage capacity (GB), weight (kg), price (yen), battery life (hours), and so on. We used these features in two ways: continuous (or original) and classified data input. For classifying the data of CPU speed, we divided the CPUs into three classes: under 1, over 1 to 2, and over 2 GHz. For classifying the data of the main memory capacity, we divided the capacities into two classes: 256 and 512 MB. For classifying the data of the HDD storage capacity, we divided the storage capacity into three classes: under 40, over 40 to 60, and over 60 GB. For classifying the data of weight, we divided the weight into five classes: under 1, over 1 to 2, over 2 to 3, over 3 to 4, and over 4 kg. For classifying the data of price, we divided the price into six classes: under 100, over 100 to 150, over 150 to 200, over 200 to 250, over 250 to 300, and over 300 thousand yen. For classifying the data of battery life, we divided the life into six classes: under 1, over 1 to 2, over 2 to 3, over 3 to 4, over 4 to 5, and over 5 hours. We inputted the data into SOM and created PC maps using SOM-Ward clustering of Viscovery SOMine 4.0 software. SOM-Ward clustering is a two-level clustering approach that combines the SOM and Ward's clustering algorithm ([Vesanto 2000], [Yao 2010]). Figures 1 and 2 show self-organizing map and an example of component map of PCs with classified data inputs, respectively. Figures 3 and 4 show self-organizing map and an example of component map of PCs with continuous (or original) data inputs, respectively. There were five clusters in Figure 1. When inspecting component maps, the feature of each cluster is clear. For example, when inspecting "under 1 GHz (1-GHz)" component map (see Figure 2), we understand that one of the features of Cluster 5 is that CPU speed is under 1 GHz. In Figure 2, originally red color (here, black) neurons correspond to under 1 GHz class and originally blue color (here, dark grey) neurons correspond to the other class.

There were four clusters in Figure 3. In CPU (GHz) component map of Figure 4, originally red (here, black) neurons correspond to 2.6 and more GHz and originally blue (here, dark grey) neurons correspond to 0.9 GHz CPU speed. Originally green and yellow (here, light grey) neurons correspond to intermediate values of CPU speed. When inspecting CPU component map of Figure 4, the feature of each cluster is not clear. So, classified data input is better than continuous (or original) data input for clustering PCs. From now, we used classified data input only. We inspected every component map and understand that features of Clusters 1 to 5 are as in Table 1.

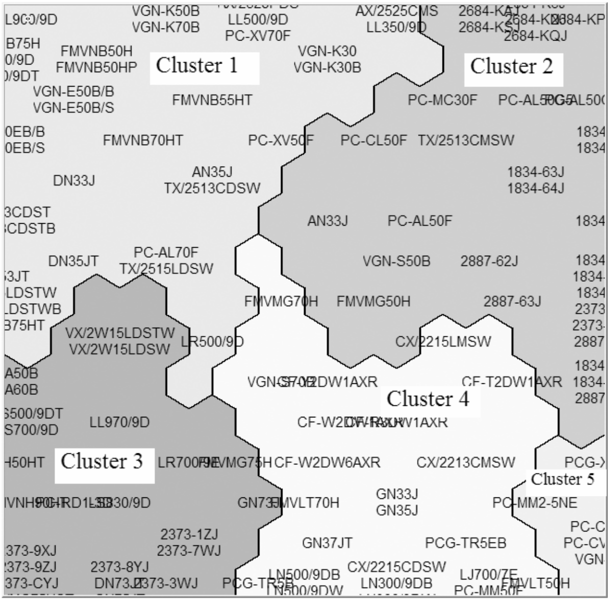

Fig. 1. Self-organizing map of PCs in 2004 with classified data inputs

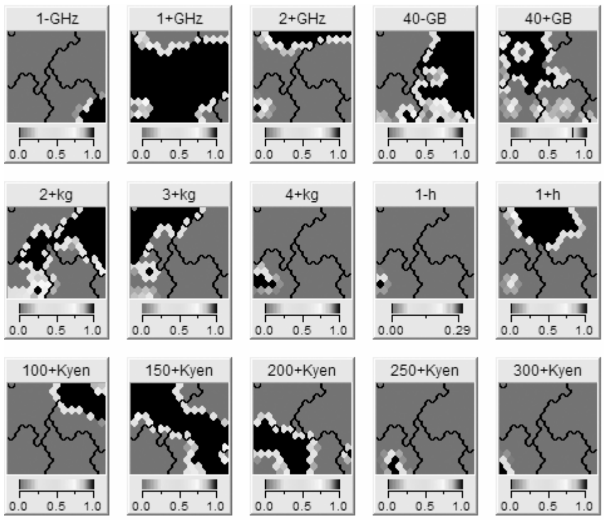

Fig. 2. Component maps of PCs in 2004 with classified data inputs

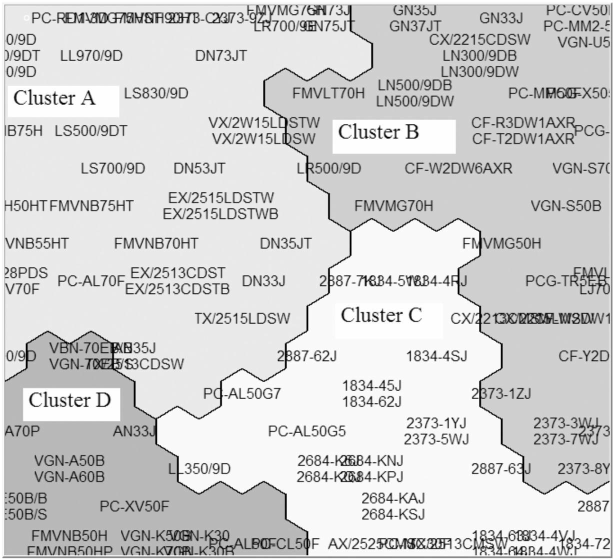

Fig. 3. Self-organizing map of PCs in 2004 with continuous data inputs

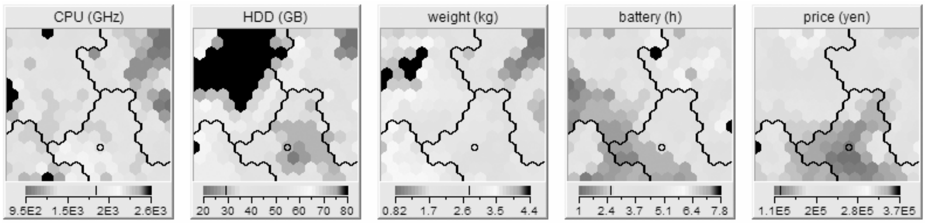

Fig. 4. Component maps of PCs in 2004 with continuous data inputs

Table 1. Main features of PCs in 2004 in each cluster with classified data inputs.

	Features	Main feature
Cluster 1	1 to 2 GHz (CPU), 40 to 60 GB (HDD), 3 to 4 Kg (weight), 150 to 200 thousand yen (price)	High performance
Cluster 2	under 40 GB (HDD), 256 MB (main memory), 100 to 150 thousand yen (price)	Low performance and low price
Cluster 3	over 60 GB (HDD), 512 MB (main memory), over 200 thousand yen (price)	Highest performance and high price
Cluster 4	1 to 2 Kg (weight), over 4 hours (battery life)	High mobility
Cluster 5	under 1 GHz (CPU), under 1 Kg (weight), 150 to 200 thousand yen (price)	Small size

Previously [Kohara 2006], we provided several class boundaries which divide input features into several classes. Because the number of classes and their boundaries depended on a person, the product maps were not always the same. In this paper, we first provide two class boundaries that divide the range between the maximum and minimum of an input feature value into three equal parts. Second, we create self-organizing product maps using the classified data inputs. We applied our way to five kinds of products (personal computers, digital cameras, automobiles, liquid crystal televisions and electronic dictionaries) and confirmed its effectiveness.

2 Creating PC Maps with SOM

The SOM algorithm is based on unsupervised, competitive learning [Kohonen 1995]. It provides a topology preserving mapping from the high dimensional space to map units. Map units, or neurons, usually form a two-dimensional lattice and thus the mapping is a mapping from high dimensional space onto a plane. The property of topology preserving means that the mapping preserves the relative distance between the points. Points that are near each other in the input space are mapped to nearby map units in the SOM. The SOM can thus serve as a cluster analyzing tool of high-dimensional data.

When we create self-organizing product maps with classified data inputs, we input 1 to the SOM if an input feature value belongs to the class. Otherwise, we input 0 to the SOM. Here, we propose a way to provide two class boundaries. We decided that the number of classes is generally three: small, middle and large. We provide two points that divide the range between the maximum (max) and minimum (min) of an input feature value into three equal parts. Two points (1/3 and 2/3 points) are calculated as follows.

1/3 point: min + (max – min)/3, 2/3 point: min + 2(max – min)/3

When we decide the max value, we ignore product data with an extremely large value (over 1.5 times the value of the 90% point) to avoid the influence of an outlier. This classification way does not mean that we remove the product data when we create self-organizing product maps.

Table 2. 1/3 and 2/3 points of PC features.

	Minimum	1/3 point	2/3 point	Maximum
CPU speed (GHz)	1.06	1.37	1.69	2
Main memory (MB)	512	683	853	1024
HDD storage (GB)	60	147	233	320
Weight (kg)	0.898	2.232	3.566	4.9
Battery life (hours)	0.9	5.6	10.3	15
Price (yen)	108,000	188,667	269,333	350,000
Monitor size (inch)	10.4	12.6	14.8	17

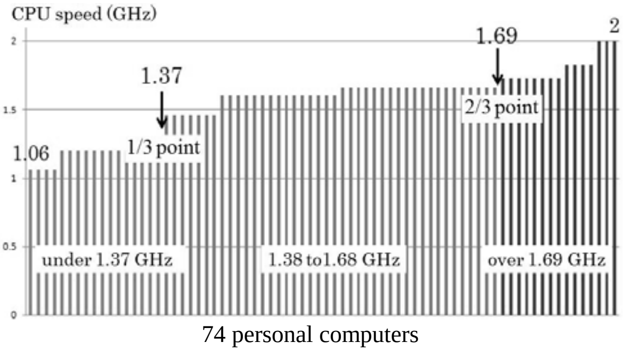

Fig. 5. Three classes of CPU speed (GHz) divided by the 1/3 and 2/3 points.

We considered 74 kinds of notebook PCs sold in Japan in 2006. We clustered these PCs according to the following features: CPU speed (GHz), main memory capacity (MB), HDD storage capacity (GB), weight (kg), price (yen), battery life (hours), and so on. For example, two class boundaries of CPU speed (GHz) are shown in Figure 5. The max and min are 2 and 1.06 GHz, respectively, and the 1/3 and 2/3 points are 1.37 and 1.69 GHz, respectively. Therefore, for the classified data of CPU speed, we divided the data into three classes: under 1.37, 1.38 to 1.68, and over 1.69 GHz. The 1/3 and 2/3 points of the PC features are shown in Table 2. We did not ignore any data to assign the 1/3 and 2/3 points of all PC features.

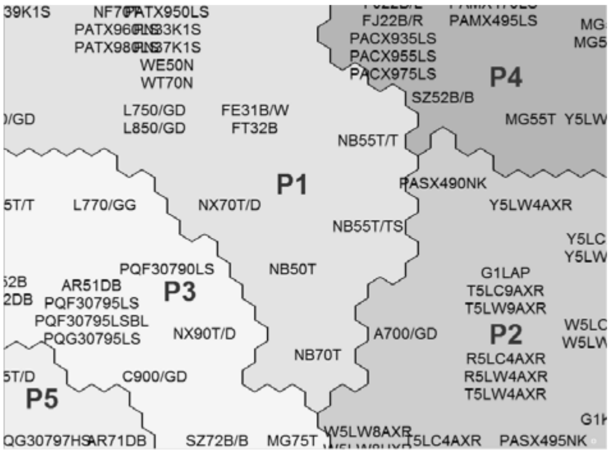

Fig. 6. Self-organizing map of PCs in 2006 with our classified data inputs.

We input each classified data to SOM and created PC maps in 2006 (Figure 6). Figure 6 shows five clusters: P1 to P5. We decided empirically that the number of clusters is around five. We adjusted the number depending on the resulting map. Examples of the component maps are shown in Figure 7. A component map shows each component value of the product map. The upper three maps of Figure 7 correspond to weight: under 2.232 kg, 2.233 to 3.565 kg and over 3.566 kg. In the "under 2.232 kg" component map, the originally red (here, black) neurons correspond to the under 2.232 kg class and the originally blue (here, dark gray) neurons correspond to the other class. Weight value of cluster P2 is under 2.232. The lower three maps correspond to battery life: under 5.6 hours, 5.7 to 10.2 hours and over 10.3 hours. "5.7 to 10.2 hours" and "10.3 hours" component maps cover cluster P2. Battery life value of cluster P2 is over 5.6 ours. When inspecting these component maps, we understand that the features of cluster P2 are light weight (under 2.232 kg) and long battery life (over 5.6 hours). Therefore, mobile PCs belong to cluster P2. We inspected every component map to understand that the features of PCs in clusters P1 to P5 are as shown in Table 3. Then, we examined whether every PC in each cluster exactly corresponds to the features, one by one. In Table 3, the underlined features are indispensable and more than 50% of the other features are necessary to judge that a PC exactly corresponds to the feature. For example, we judge a PC in cluster P2 as exactly mobile when its weight is under 2.232 kg *and* its battery life is over 5.6 hours. We judge a PC in cluster P4 as having low performance and low price when its price is under 188,667 yen *and* its main memory size is under 683 MB *or* its HDD size is under 147 GB. We judge a PC in cluster P1 as having high CPU speed *and* heavy weight when its CPU speed is over 1.38 GHz *and* its weight is over 2.233 kg. The accuracy of each cluster is also shown in Table 3. The total accuracy is 95.9%.

Table 3. Main features of PCs in 2006 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster P1 (25)	<u>over 1.38 GHz</u> (CPU), <u>over 2.233 kg</u> (weight)	High CPU speed and heavy weight	25/25
Cluster P2 (18)	<u>under 2.232 kg</u> (weight), <u>over 5.6 hours</u> (battery life)	High mobility	18/18
Cluster P3 (13)	<u>over 1.38 GHz</u> (CPU), over 853 MB (main memory), over 148 GB (HDD), over 14.8 inches (monitor size)	High performance	11/13
Cluster P4 (15)	under 683 MB (main memory), under 147 GB (HDD), <u>under 188,667 yen</u> (price)	Low performance, low price	14/15
Cluster P5 (3)	<u>over 1.69 GHz</u> (CPU), <u>over 853 MB</u> (main memory), <u>over 269,333 yen</u> (price)	Highest erformance and high price	3/3
Total (74)		71/74 = 95.9%	

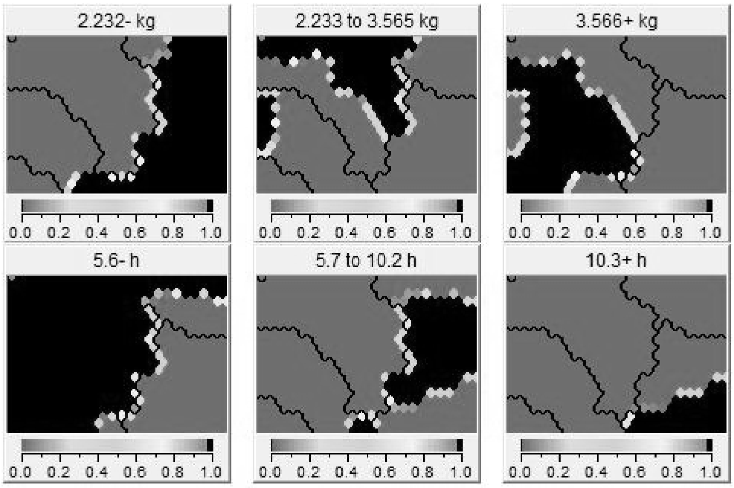

Fig. 7. Examples of component maps of PCs in 2006 with our classified data inputs.

3 Applying Our Way to Other Products

We applied our way to other products sold in Japan: 123 digital cameras in 2006, 142 automobiles in 2006, 60 liquid crystal TVs in 2006, 67 electronic dictionaries in 2008 and 86 recent PCs in 2009 (Windows 7 Home Premium). When deciding the max value, we ignored a total of 0.8% of the product data, which was data containing an extremely large value (over 1.5 times the value of the 90% point), as shown in Table 4. Self-organizing product maps are shown in Figures 8 to 12. The main features and accuracy are shown in Tables 5 to 9, where the underlined features are indispensable and more than half of the other features are necessary. The total accuracy is 97.6% for digital cameras, 95.8% for automobiles, 95.0% for liquid crystal TVs, 94.0% for electronic dictionaries and 96.5% for recent PCs. Therefore, we confirmed the effectiveness of our way.

Table 4. Percentage of ignored product data when deciding the max value.

Product	# of product data	# of ignored data	% of ignored data
PCs in 2006	518	0	0
Digital cameras	882	11	1.2
Automobiles	568	8	1.4
Liquid crystal TVs	240	1	0.4
Electronic dictionaries	335	2	0.6
PCs in 2009	602	4	0.7
Total	3,145	26	0.8

# of product data = (number of products) times (number of continuous features)

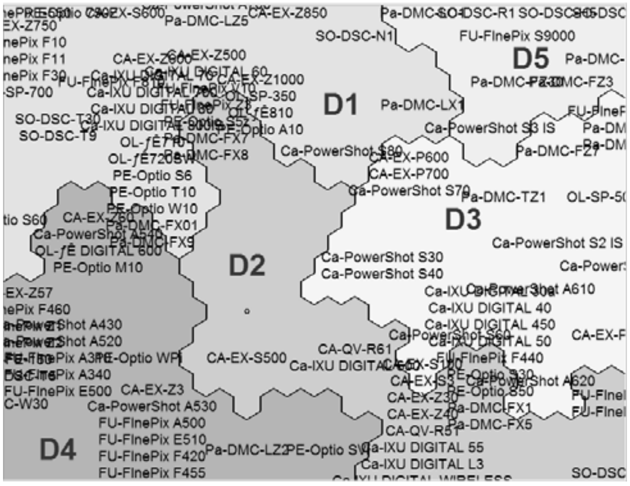

Fig. 8. Self-organizing map of digital cameras in 2006 with our classified data inputs.

Table 5. Main features of digital cameras in 2006 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster D1 (34)	over 5,430 thousand (# of pixels)	Large number of pixels	34/34
Cluster D2 (36)	under 2.5 inch (monitor size), under 221.3 g (weight)	Small size and light weight	36/36
Cluster D3 (17)	under 5,430 thousand (# of pixels), over 221.3 g (weight)	Small number of pixels, heavy weight	17/17
Cluster D4 (30)	under 29,733 yen (price)	Low price	29/30
Cluster D5 (9)	over 10 times (optical zoom)	10 times optical zoom	7/9
Total (126)		123/126 = 97.6%	

Table 6. Main features of automobiles in 2006 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster A1 (66)	under 2,205,000 yen (price), under 1.437 t (weight), under 1,607 cc (emissions)	Low price, light weight and low emissions	62/66
Cluster A2 (31)	1.438 to 2.172 t (weight), 1,608 to 2,552 cc (emissions), under 13.0 km/l (efficiency)	Middle weight, middle emissions, fuel-inefficient	31/31
Cluster A3 (23)	over 13.0 km/l (efficiency)	Fuel-efficient	22/23
Cluster A4 (22)	over 2,205,000 yen (price)	High price	21/22
Total (142)		136/142 = 95.8%	

Fig. 9. Self-organizing map of automobiles in 2006 with our classified data inputs.

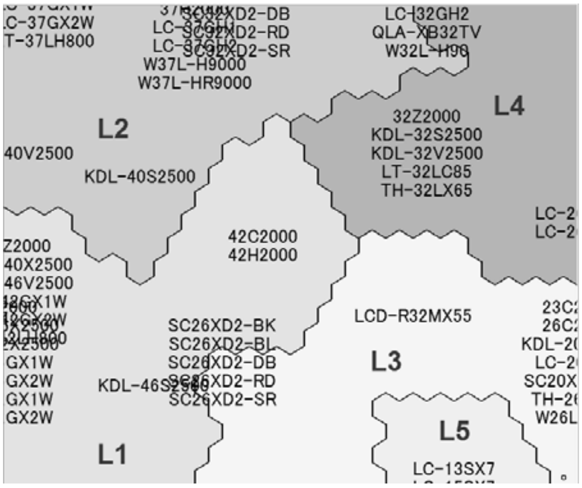

Fig. 10. Self-organizing map of liquid crystal TVs in 2006 with our classified data inputs.

Table 7. Main features of liquid crystal TVs in 2006 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster L1 (16)	<u>over 39 inches</u> (monitor), <u>over 232 W</u> (power)	Largest size and high power consumption	16/16
Cluster L2 (17)	<u>27 to 38 inches</u> (monitor), <u>137 to 231 W</u> (power)	Middle size and middle power consumption	15/17
Cluster L3 (13)	<u>under 26 inches</u> (monitor), <u>under 136 W</u> (power)	Small size and low power consumption	12/13
Cluster L4 (12)	<u>under 162,917 yen</u> (price), <u>137 to 231 W</u> (power)	Low price and middle power consumption	12/12
Cluster L5 (2)	<u>under 162,918 yen</u> (price), <u>under 27 inches</u> (monitor), <u>under 137 W</u> (power)	Low price, smallest size and low power consumption	2/2
Total (142)			57/60 = 95.0%

Table 8. Main features of electronic dictionaries in 2008 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster E1 (19)	18,581 to 32,538 yen (price), 196 to 277 g (weight), over 4.93 inches (monitor), under 37 (dictionaries)	Middle price, middle weight, large size, few dictionaries	19/19
Cluster E2 (20)	over 22,550 yen (price), over 4.93 inches (monitor)	High price and large size	17/20
Cluster E3 (7)	under 4.37 inches (monitor)	Small size	7/7
Cluster E4 (7)	under 18,580 yen (price), 4.38 to 4.92 inches (monitor)	Low price and middle size	3/4
Cluster E5 (8)	over 4.93 inches (monitor), over 68 (dictionaries),	Large size and many dictionaries	8/8
Cluster E6 (9)	under 18,580 yen (price), under 4.37 inches (monitor), under 37 (dictionaries)	Low price, small size and few dictionaries	9/9
Total (67)		63/67 = 94.0%	

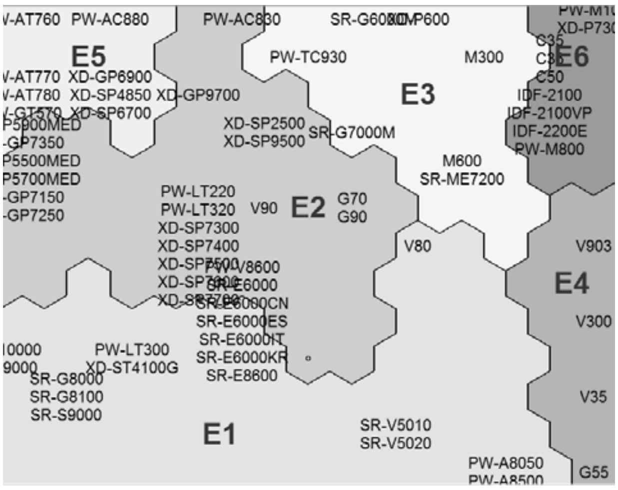

Fig. 11. Self-organizing map of electronic dictionaries in 2008 with our classified data inputs.

Table 9. Main features of PCs in 2009 in each cluster.

Cluster # (# of products)	Features	Main feature	Accuracy
Cluster N1 (24)	1.74 to 2.26 GHz (CPU), 248 to 372 GB (HDD), 2.34 to 3.46 kg (weight)	Middle performance, middle weight	24/24
Cluster N2 (25)	under 1.73 GHz (CPU), under 2.33 kg (weight), under 99,959 yen (price)	Low performance, light weight, low price	25/25
Cluster N3 (21)	over 2.27 GHz (CPU), over 3 GB (main memory)	High performance	20/21
Cluster N4 (11)	over 5.0 hours (battery life), under 2.33 kg (weight)	High mobility	10/11
Cluster N5 (5)	over 2.27 GHz (CPU), over 3 GB (main memory), over 15.6 inches (monitor), over 154,158 yen (price), Blu-ray drive	Highest performance and high price	4/5
Total (86)		83/86 = 96.5%	

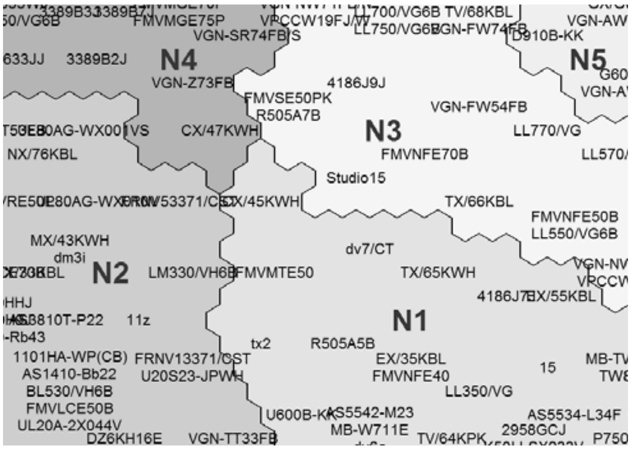

Fig. 12. Self-organizing map of PCs in 2009 with our classified data inputs.

4 Purchase Decision Making with AHP

AHP is a multi-criteria decision method that uses hierarchical structures to represent a problem and to develop priorities for alternatives based on the user. [Saaty 1980] has shown that weighting activities in multi-criteria decision making can be effectively dealt with via hierarchical structuring and pairwise comparisons. Pairwise comparisons are based on forming a judgment between two particular elements rather than attempting to prioritize an entire list of elements. There are five types of AHP: relative measurement, absolute measurement, inner dependence, outer dependence and inner-outer dependence [Saaty 1980]. There are two types of ANP (Analytic Network Process): feedback system and series system [Saaty 1996]. We can choose an appropriate type of AHP or ANP according to the property of a problem. Therefore, we choose AHP for decision making. The AHP scales of pairwise comparisons are shown in Table 10.

Table 10. The AHP scales for pairwise comparisons.

Intensity of importance	Definition and explanation
1	Equal importance
3	Moderate importance
5	Essential or strong importance
7	Demonstrated importance
9	Extreme importance
2, 4, 6, 8	Intermediate values between the two adjacent judgments when compromise is needed.

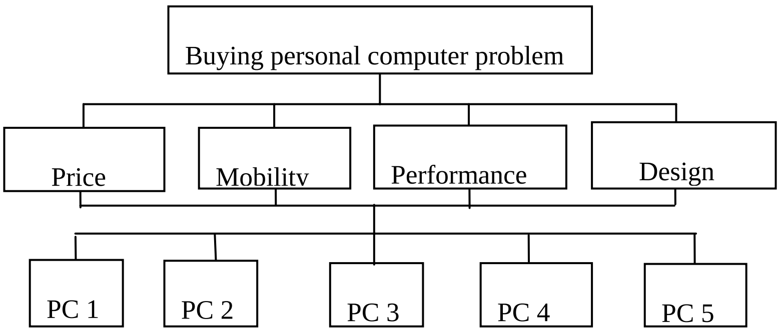

Fig. 13. AHP model for problem of buying a PC.

Figure 13 shows the relative measurement AHP model for the problem of buying a PC. For the goal on the first level (i.e., the problem of buying a PC), four criteria on the second level and five alternatives on the third level were defined. Here, we used the following four criteria: low price, high mobility, high performance, and preference of design. Here, high mobility means light weight and long battery life. High performance means high CPU speed, large main memory capacity, large HDD storage capacity and large monitor. We can select some alternatives using the PC maps in several ways: from a favorite cluster, from a favorite component map, from a favorite brand, from a total map. For example, we selected five alternatives (Table 14) using the *neighborhood view* function of Viscovery SOMine 4.0 software (this function displays all neurons that are topologically similar to a reference neuron) from a favorite cluster N3 of a recent PC map whose main feature is high performance, as shown in Figure 14. Here, PC 1 (FMVNFE70B) is a favorite PC and a reference neuron. We referred PC 1 and obtained four neighbor PCs using the maps.

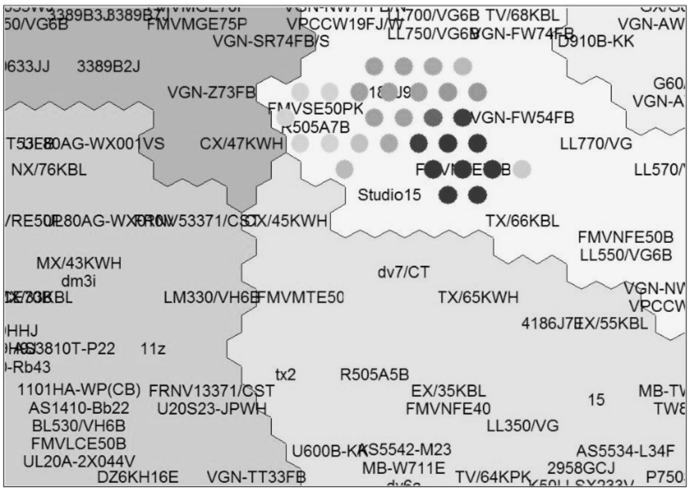

Fig. 14. Selection of alternatives using neighborhood view function.

Then, we applied AHP. The pair comparison matrix among four criteria considered by one of us is shown in Table 11. For example, price is strongly important in comparison to mobility. Performance is strongly important in comparison to design. As a result, performance is most important (its weight = 0.515). Consistency index means whether a pair comparison matrix is consistent or not. When the index is lower than 0.10, we judge that the pair matrix is consistent [Saaty 1980]. When the index is larger than 0.10, pairwise comparisons should be reconsidered. The pair comparison matrix for price is shown in Table 12. From the point of view of low price, PC 3 is strongly important in comparison to PC 1 and PC 2. The weight of PC 3 was highest (its weight = 0.562). The weight matrix for four criteria is shown in Table 13. We obtain final results as follows: final results = the weight matrix for four criteria (Table 13) times the weight matrix among four criteria (Table 11). For example, the result for PC 3 is obtained as follows.

0.562 * 0.293 + 0.09 * 0.050 + 0.222 * 0.515 + 0.109 * 0.142 = 0.299

In this case, performance is the most important and price is important. Because PC 3 is comparatively low price, PC 3 is selected as the final choice (Table 14).

Table 11. Pair comparison matrix among four criteria.

	Price	Mobility	Performance	Design	Weight
Price	1	5	1/2	3	0.293
Mobility	1/5	1	1/7	1/5	0.050
Performance	2	7	1	5	0.515
Design	1/3	5	1/5	1	0.142

Consistency index = 0.064

Table 12. Pair comparison matrix for price.

	PC 1	PC 2	PC 3	PC 4	PC 5	Weight
PC 1	1	1/2	1/5	3	3	0.125
PC 2	2	1	1/5	5	5	0.208
PC 3	5	5	1	7	7	0.562
PC 4	1/3	1/5	1/7	1	2	0.060
PC 5	1/3	1/5	1/7	1/2	1	0.045

Consistency index = 0.066

Table 13. Weight matrix for four criteria.

	Price	Mobility	Performance	Design
PC 1	0.125	0.056	0.222	0.369
PC 2	0.208	0.373	0.111	0.206
PC 3	0.562	0.090	0.222	0.109
PC 4	0.060	0.108	0.222	0.206
PC 5	0.045	0.373	0.222	0.109

Table 14. Alternatives and final results of AHP for problem of buying a PC.

	CPU (GHz)	Mem. (GB)	Monitor (inches)	Weight (kg)	Battery (hours)	Price (yen)	Results
PC 1	2.53	4	15.6	2.80	2.1	129,800	0.206
PC 2	2.53	4	14.1	2.50	3.9	122,280	0.166
PC 3	2.53	4	15.4	2.70	2.4	109,800	0.299
PC 4	2.53	4	16.4	3.20	3.0	141,871	0.167
PC 5	2.66	4	15.6	2.75	4.0	148,799	0.162

5 Conclusion

We propose a way of creating product maps with SOM. First, we provide two class boundaries which divide the range between the maximum and minimum of an input feature value into three equal parts. Second, we create self-organizing product maps using the classified data inputs. We applied our way to five kinds of products. For all the products, we confirmed the effectiveness of our way. In future work, we will apply our way to other products, very large problems (for example, the number of products is very large) and other real-world clustering problems. We will investigate a way of improving the accuracy of each cluster. We will use other types of AHP and ANP for decision making.

References

Ha, L. et al.: Facilitating Electronic Business Planning with Decision Making Support Systems. In: Palade, V. et al. (Eds.): Knowledge-Based Intelligent Information and Engineering Systems, LNAI, vol. 2774, pp. 45-51, Springer-Verlag, Heidelberg (2003)

Kohara, K.: Neural Networks for Economic Forecasting Problems. In: Leondes, C. T. (Ed.): Expert Systems - The Technology of Knowledge Management and Decision Making for the 21st Century. Academic Press, San Diego (2002)

Kohara, K., Isomae, M.: Purchase Decision Support with Self-Organizing Maps and Analytic Hierarchy Process. In: Vale, Z. et al., (Eds.): Proc. International Conf. on Knowledge Engineering and Decision Support, pp. 151-157, Lisbo. (2006)

Kohonen, T.: Self-Organizing Maps. Springer, New York (1995)

Kotler, P.: Marketing Management, eleventh edition. Prentice Hall, New Jersey (2002)

Park, C.-S., Han, I.: A CBR with feature weights derived by analytic hierarchy process for bankruptcy prediction. Expert Systems with Applications 14(3), 255-264 (2002)

Riordan, D., Hansen, B.K.: A fuzzy case-based system for weather prediction. Engineering Intelligent Systems 3, 139-145 (2002)

Saaty, T.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)

Saaty, T.: The Analytic Network Process. Expert Choice, Arlington (1996)

Suka, M. et al.: Clinical Decision Support System Applied the Analytic Hierarchy Process. In: Palade, V. et al. (Eds.): Knowledge-Based Intelligent Information and Engineering Systems, LNAI, vol. 2774, pp. 417-423, Springer-Verlag, Heidelberg (2003)

Vesanto, J., Alhoniemi, E.: Clustering of the Self-Organizing Map. IEEE Transactions on Neural Networks 11 (3), 586-600 (2000)

Walle, B. V., Moldovan, M.: An Information Market for Multi-agent Decision Making: Observation from a Human Experiment. In: Palade, V. et al. (Eds.): Knowledge-Based Intelligent Information and Engineering Systems, LNAI, vol. 2774, pp. 66-72, Springer-Verlag, Heidelberg (2003)

Yao, Z. et al.: Combining Unsupervised and Supervised Data Mining Techniques for Conducting Customer Portfolio Analysis. In: Perner. P. (Ed.): Advances in Data Mining, LNAI, vol. 6171, pp. 292-306, Springer-Verlag, Heidelberg (2010)

Vitae

Kazuhiro Kohara received Bachelor and Master degrees in Applied Physics from the University of Tokyo in 1976 and 1978, respectively. He joined NTT Laboratories in 1978. He received Ph.D. degree in Computer Science from the University of Tokyo in 1997. He was a visiting research assistant professor of Department of Computer Science, University of Illinois at Urbana-Champaign. Currently, he is a professor of Chiba Institute of Technology. His research interests are machine learning, data mining, text mining, decision making, multi-agent simulation, self-organizing maps, genetic algorithm and natural language processing.

Tetsuya Tsuda received Bachelor degree in Engineering from Chiba Institute of Technology in 2009. Currently, he is a student of Graduate school, Chiba Institute of Technology. His research interests are self-organizing maps and decision making.