CHAPTER 2

Experimental and methods

2.1 Chemical, Apparatus and Instruments

2.1.1 Chemicals

1) Rice sample

2) 1.00 µL of 0.50 mg/mL 2, 6-Dimethoxyphenol (DMP) in toluene

3) 2,4,6-Trimethylpyridine (TMP)

2.1.2 Apparatus and instruments

1) Powdered machine; Cyclotec TM 1093

2) The 0.5 mm screen sieve; Foss, Hoganan, Sweden

3) Gas chromatograph; BRUKER technologies model 450-GC

4) GC column; HP-5MS capillary (dimension of 30 m × 0.32 mm i.d. × 0.25 µm film thickness)

5) Headspace sampler; QUMA headspace sampler model 40/111

6) Nitrogen-phosphorus detector (NPD)

2.2 Design of experiment (DOE)

In this work, the experiment was systematically designed in order to provide sufficient information from the smallest number of experiments. Central composite design (CCD), a method of DOE, was chosen because it comprises the advantages of full a factorial design, a star design and replicates at the centroid [18, 29]. This experiment has five treatment levels in each factor. The relationships between independent factors and response factor were investigated using mathematical model that was generated using the experimental data from CCD. Moreover predictive ability of the response could be provided using the established model.

In this work, the concentrations of nitrogen fertilizer and sodium salt were varied in each experiment. There were 9 treatments corresponding to 9 combinations of the studied factors. Each of the treatments had 5 replicate pots; therefore, the constructed design consisted of 45 experimental rice plants in total. Each of the rice plants was hydroponically grown in a plastic pot (10 L; 20 cm high). The rice cultivar was Pathum Thani 1 (PT1), an improved fragrant temperate *indica* type (*Oryza sativa* L.) and non-photosensitive rice [16]. The nutrient solution used in this studied was modified based on Hoagland’s nutrient standard solution [30]. Ammonium nitrate (NH_4NO_3) was used as the nitrogen source, while sodium chloride (NaCl) was used to induce saline condition in the experiments. The concentration of each factor was used in the range of maximum and minimum concentration that rice plant could be successfully grown. The concentrations of nitrogen and sodium salt in nutrient solutions were shown in Table 2.1.

Prior to the transplanting, the rice seeds were soaked in water for 24 h, and then incubated under moist and dark condition for 48 h. After that, each of the germinated seeds was transplanted in a sponge placed inside 1.5 inch thick plastic foam floating on the nutrient solution. During the first week of the transplanting, 50% concentration of the standard Hoagland’s nutrient solution was used. Later, each of the experimental pots was treated with the nutrient solution according to the experimental design. The nutrient solutions were renewed weekly and the water level for each of the pots was maintained daily using fresh water if necessary.

Table 2.1 Coded values and the concentrations of N and NaCl added to the standard Hoagland's nutrient solution for the central composite design (CCD)

Treatments	Coded Values		Concentrations (ppm)	
	N	NaCl	N	NaCl
1	0.71	0.71	234	34
2	0.71	-0.71	234	6
3	-0.71	-0.71	150	6
4	-0.71	0.71	150	34
5	1	0	252	20
6	-1	0	132	20
7	0	1	192	40
8	0	-1	192	0
9	0	0	192	20

The experiments were conducted in Mae Hia Agricultural Research, Demonstrative and Training Center, Chiang Mai University in Chiang Mai, Thailand, during the dry season (November 2013 to March 2014). The plant growth parameters including of number of tillers, plant height and root length were measured at various times (Table 2.2). After harvest, yield components such as number of grains per panicle, panicle length, plant weight, shoot and root dry weights, number of panicles per plant, number of grains per plant and thousand grain weight were collected (Table 2.3).

Table 2.2 Growing data of the rice plants [a]

Treat.	Transplanting			Tillering			Panicle initiation			Harvest	
	Plant height (cm)	Tillers	Root length (cm)	Plant height (cm)	Tillers^b	Root length^b (cm)	Plant height (cm)	Tillers^b	Root length^b (cm)	Plant height^b (cm)	Tillers
1	7.23 ± 1.26	3 ± 1	13.57 ± 2.50	7.80 ± 1.87	6 ± 2	17.33 ± 4.54	19.87 ± 2.51	16 ± 5	31.33 ± 2.47	73.33 ± 3.14	114 ± 31
2	6.00 ± 0.83	3 ± 1	16.54 ± 4.28	8.92 ± 0.58	8 ± 1	18.80 ± 2.61	19.70 ± 1.57	39 ± 6	27.00 ± 3.92	81.41 ± 4.79	117± 21
3	6.63 ± 0.91	3 ± 0	17.12 ± 1.23	8.50 ± 0.60	9 ± 1	18.88 ± 1.55	18.58 ± 1.59	55 ± 6	26.00 ± 1.08	84.76 ± 1.80	100 ± 11
4	7.28 ± 0.83	4 ± 1	15.80 ± 1.96	8.92 ± 0.32	9 ± 1	20.60 ± 2.43	18.80 ± 0.97	27 ± 4	30.00 ± 2.57	81.52 ± 3.57	94 ± 16
5	6.76 ± 1.18	3 ± 1	16.42 ± 4.20	8.24 ± 1.10	9 ± 2	19.00 ± 4.06	19.34 ± 0.76	47 ± 13	26.00 ± 2.00	85.32 ± 2.53	115 ± 17
6	7.10 ± 0.50	3 ± 1	18.58 ± 2.16	8.20 ± 0.56	9 ± 1	23.00 ± 1.00	18.48 ± 1.66	29 ± 7	34.20 ± 2.11	80.67 ± 4.57	99 ± 21
7	7.40 ± 1.16	3 ± 0	16.64 ± 2.54	9.30 ± 0.78	7 ± 2	21.28 ± 1.13	19.92 ± 0.79	27 ± 7	31.50 ± 3.06	79.22 ± 7.95	93 ± 16
8	7.42 ± 1.77	3 ± 0	16.88 ± 3.82	9.10 ± 2.16	10 ± 2	19.62 ± 4.82	19.90 ± 1.68	62 ± 11	23.88 ± 2.32	90.21 ± 3.83	112 ± 12
9	7.66 ± 0.59	4 ± 1	18.74 ± 2.30	9.38 ± 0.83	11 ± 1	23.10 ± 2.46	19.14 ± 0.86	45 ± 3	25.10 ± 1.29	85.12 ± 3.76	105 ± 12

^{a}Data are means ± SD. ^{b}Statistically significant at $p < 0.05$.

Table 2.3 Yield component data[a]

Treat.	Grains per panicle	Panicle length (cm)	Plant weight[b] (kg)	Shoot dry weight[b] (g)	Root dry weight (g)	Panicles per plant	Grains per plant	1000 grains weight (g)
1	127 ± 21	26.33 ± 1.41	2.54 ± 0.66	444.60 ± 26.84	43.01 ± 4.94	116 ± 16	1285 ± 512	20.84 ± 0.99
2	127 ± 19	25.67 ± 0.95	2.75 ± 0.09	552.20 ± 51.48	48.87 ± 8.44	118 ± 16	3631 ± 1627	21.68 ± 0.79
3	143 ± 16	26.89 ± 1.84	2.50 ± 0.22	505.53 ± 20.27	50.85 ± 6.16	107 ± 17	3095 ± 2005	23.28 ± 0.55
4	142 ± 5	27.41 ± 1.31	1.94 ± 0.18	403.69 ± 57.24	43.69 ± 4.36	96 ± 12	3297 ± 728	21.57 ± 1.02
5	133 ± 12	26.70 ± 1.73	2.36 ± 0.31	460.37 ± 84.20	42.46 ± 7.70	118 ± 13	3664 ± 1312	22.16 ± 1.08
6	147 ± 15	26.72 ± 1.23	2.20 ± 0.15	428.51 ± 40.56	47.45 ± 5.11	114 ± 21	4036 ± 1493	22.30 ± 1.29
7	142 ± 31	26.49 ± 2.24	2.06 ± 0.28	393.42 ± 75.07	41.71 ± 5.10	108 ± 16	3364 ± 1092	21.34 ± 1.20
8	145 ± 10	26.42 ± 1.38	2.74 ± 0.31	524.40 ± 134.26	49.60 ± 8.81	120 ± 12	3845 ± 1511	24.19 ± 0.42
9	134 ± 21	26.46 ± 1.52	2.17 ± 0.24	378.21 ± 58.90	43.28 ± 5.43	102 ± 16	2971 ± 1614	21.35 ± 0.82

[a] Data are means ± SD. [b] Statistically significant at $p < 0.05$.

2.3 Determination of 2-acetyl-1-pyrroline (2AP) in rice grains

The nine rice grain samples, from mixing rice samples of each replicate, that represent nine experimental samples was powdered using a Cyclotec TM 1093 mill with a 0.5 mm screen sieve (Foss, Hoganan, Sweden). One gram (1.00 g) of each sample was then put into a 20 mL headspace vial, this being followed by the addition of 1.00 µL of 0.50 mg/mL 2, 6-DMP in toluene. The headspace vial was then immediately sealed with a PTFE/silicone septum and aluminum crimp cap. It was shaken at a room temperature of 27 °C for 10 min prior to analysis by SHS-GC.

SHS-GC determination of 2AP was carried out using BRUKER technologies (Goes, Netherlands) model 450-GC gas chromatograph equipped with an QUMA headspace sampler model 40/111 (Wuppertal, Germany). A nitrogen-phosphorus detector (NPD) was used for selective determination of the aroma impact compound, 2AP. Optimization of SHS-GC-NPD condition was performed similar to the way described by Sriseadka *et al*. [31]. The loop filling time, pressurizing time, and injection time were set at 0.01, 0.50, and 0.50 min, respectively. Oven temperature was 120 °C, vial equilibration time at 10 min, and sample loop and transfer line temperatures were set at 130 and 135 °C, respectively. A sample headspace was collected through a 1 mL sample loop and automatically transferred to the GC via a heated transfer line. The GC column was a HP-5MS capillary with dimension of 30 m × 0.32 mm i.d. × 0.25 µm film thickness (J&W Scientific, Folsom, CA) and was operated in a temperature program mode that started at 50 °C and then increased 5 °C/min to 125 °C. The NPD temperature was 300 °C. SHS-GC-NPD employed a splitless injection at 230 °C and carrier gas flow rate was 2 mL/min.

2AP contents in rice seed samples were determined by means of a standard calibration curve. Areas under peaks of 2AP and 2,4,6-Trimethylpyridine (TMP), used as internal standard, were measured and the ratios were correlated with concentrations of 2AP in the rice samples. The calibration curve was set to be linear over the concentration range of 0.10–10.00 mg/L for 2AP with a regression coefficient of 0.996. The relative standard deviation calculated for each data point of concentration was less than 10%, based on 3 independent runs.

2.4 Chemometric analyses

In this work, the studied factors should be simultaneously used to predict 2AP content and investigated their effect to response factor (2AP content in rice grain). Due to the traditional method could not be used, chemometric techniques were chose. Chemometrics were used for the two proposes, establish model to predict 2AP content and investigate the effect of each studied factors on 2AP content. Partial least squares (PLS) regression was used to establish the model to predict 2AP content. Moreover, the PLS coefficients showed the effect of each studied factor on 2AP content. Supervised self-organizing map (SSOM) was used to study the behavior of studied factors including to response factor.

2.4.1 Partial least squares (PLS) regression

Partial least squares (PLS) regression is one of the most powerful multivariate calibration methods. This method captures variations from both of the predictive (X) and response (c) parameters and these variations were simultaneously used for constructing the regression model [22]. Therefore, PLS, in most cases, could satisfactorily provide predictive results. In this work, the design data, plant growth data and yield-related components were used as the predictive parameters where the concentrations of 2AP in the rice grains were used as a response. A simple approach to analyze these data was to generate individual PLS model where the design data was only used as predictor. The established models could be used to investigate the effect of the design data on the 2AP content. Also, it was possible to investigate how the design data affected the plant growth parameters or the yield-related components. However, this was not easy to envision the effect of these additional parameters with respect to the aromatic quality of the rice grains. The aim of this research was to investigate if it was possible to estimate the rice aromatic quality based on the designed parameters. At the same time, we sought to identify and interpret any other factors influencing the rice aromatic quality. Therefore, these parameters were additionally incorporated into the predictive parameters for the PLS model resulting in a data with multiple blocks.

In fact, there are several algorithms that could be used to cope with the multiblock data. For example, multiblock PLS (MB-PLS) [32] and serial PLS (S-PLS) [33]. Using MB-PLS, weighting factors between the data blocks were identified as a parameter affecting on the model prediction [34]. The MB-PLS solution was the same as ordinary PLS for the same weighting of the data blocks. Placing different weights on data, in this case the design data versus the plant growths and the component yields, could have impact on the predictive results but this could also blur interpretation of the relationship between the parameters. Moreover, MB-PLS required that the same number of components were applied for all the data blocks which, in this case, was limited to 2 according to the number of design parameters used. On one hand, S-PLS differs from MB-PLS in that S-PLS decomposed the predictor blocks in a serial mode, and therefore, it was possible to use a different number of components for the different data blocks. But, if the model acquired enough information from the first block data for modeling the response, only small improvement could be achieved when the second block data was used. Consequently, the coefficients of the second block data calculated using S-PLS were relatively smaller. This made S-PLS impossible to see if there were variables in the second block that might contribute to the model. An alternative analysis for these situations was to use an ordinary PLS. In this case, the data blocks were simply concatenated into a single block and a conventional PLS was performed. Several works have reported that an ordinary PLS with the combined data seemed to perform rather the same as the modified multiblock algorithms regarding the variance of the parameter estimates [34, 35]. Yaroshchyk *et al*. [36] reported that, in their work, the ordinary PLS performed the best when compared with the multiblock regressions (MB-PLS and S-PLS).

The optimum number of PCs of the PLS model was determined using leave-one-out cross-validation (LOOCV) [19]. In addition to the predictive model, the significance of each of the parameters was evaluated using PLS coefficients and variable influence on projection (VIP) [37]. For a calibration model, different data pre-processing can strongly influence the analysis results and there are no clear-cut guidelines when to use or to avoid certain pre-processing methods. Based on a "trial-and-error" approach, several data pre-processing methods and the order in which the methods were applied were tested. The outcome was that square root scaling following by mean centring resulted in the best performing model. The model with a unit-variance scaling or standardization were also tested, but showed less satisfactory results. Therefore, for this dataset, square root scaling with mean centring were chosen. In most case, standardization could equalize the effect of each variable to contribute to the model evaluation. But, unimportant variables which were not useful for the prediction became important and they could perhaps worsen the model prediction. On the other hand, a square root transformation, though not completely, could reduce influences of variables with large values brining about a "pseudo-scaling" effect. In addition, mean centring adjusted all the parameter values to fluctuate around zero instead of around their means. Hence, it adjusted for the difference in the offset between high and low sizes of parameters, and therefore, could be used to focus on the variation parts of the data for the analysis.

2.4 2 Empirical method**:** Monte Carlo for confirming the significance test

To confirm the significance of the parameters picking out by the PLS coefficients, a Monte Carlo experiment was performed [19]. This empirical significance test was investigated to determine how large the PLS coefficients obtained from the certain data by comparing with the coefficients obtained during Monte Carlo permutations or a set of null distribution. This null distribution was created using randomly permuted vector of the response, in this case, the 2AP contents. Then, the PLS model was constructed based on the randomly assigned 2AP vector resulting in another set of PLS coefficients. This procedure was repeated for 5000 times; therefore, generating a null distribution for benchmarking the significance of the coefficient for each parameter. It is noted that the distribution of the coefficients was two tailed, hence the absolute PLS coefficients were used for the comparison. In this work, a threshold of 90% was used. If a coefficient was higher than this value, it was considered to be a significant parameter.

2.4.2 Self-organizing map (SOM)

Self Organising Maps (SOMs) were first described by Kohonen in the 1980s as a method for visualising the relationship between different speech patterns [38]. SOMs can be considered as an unsupervised learning method which is an alternative to principal component analysis (PCA) since they can be used to present the structure of a data using a low-dimensional display, although it also can be modified to be used in a supervised mode. A SOM involves a map which is often represented by a two dimensional grid consisting of a certain number of units. This map will be trained using training samples and the aim is to locate the positions of the training samples on the map such that the relative distance between them in the original data space is preserved as much as possible. SOM algorithm has several stages and a number of parameters that need to be set, as described below.

- Initialisation

In the initialisation process, a trained map consisting of a grid of units is generated. The shape of the units is not specific, although squares and hexagonal are particularly favourable because they have neighbours that have the same distance apart in numerous directions (Figure 2.1).

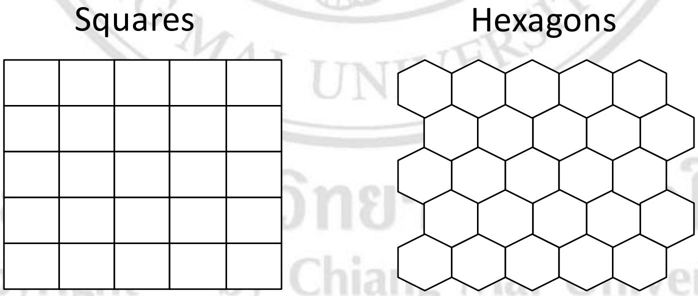

Figure 2.1 Example of lattices used for SOMs

Yet, the difference between squares and hexagons is that the squares favour only the vertical and horizontal directions during the learning process, whereas the hexagons do not. In this chapter, the SOM algorithm is demonstrated using hexagonal units, however, the same algorithm can be adopted when the other shapes are used. A map consisting of K ($=P\times Q$) map units in the hexagons is shown in Figure 2.2. Each map unit is characterised by a weight for each variable in the dataset, resulting in a $1\times J$ weight vector $\boldsymbol{w}_k$, where J corresponds to the number of variables. Therefore, each variable *j* has *K* weight units which, in this thesis, are generated from randomly selected values from a uniform distribution within the measured range of variable *j*.

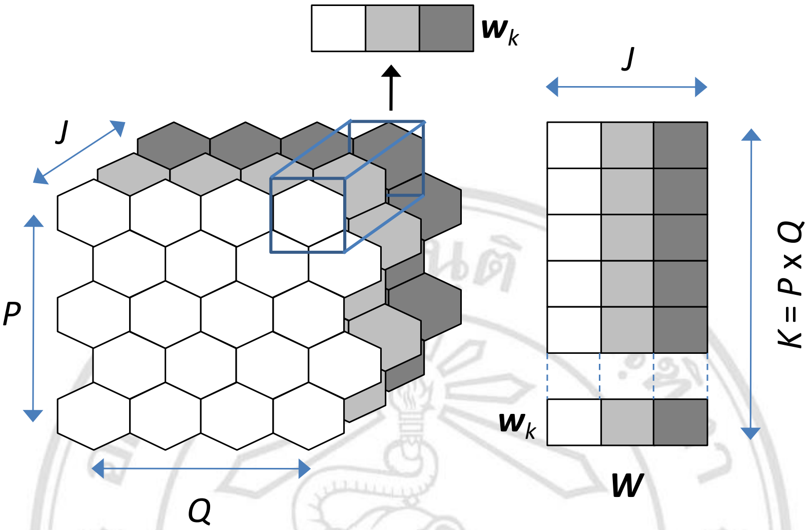

Figure 2.2 $P \times Q$ map with J weights containing a total of K map units and the corresponding weight matrix $\boldsymbol{W}$.

- Training

After the map containing random values for each variable is created, this map will be trained. During the training process, the samples gradually move towards a region of the map that they are most similar to, and so samples that are close together in the high-dimensional input space are mapped to the units that are close together in the map space. Two important parameters, the neighbourhood width φ and the learning rate ψ, control this training process. Both of them are dependent on the dimensions of the map and the total number of iterations as described below.

- Neighbourhood Width

The neighbourhood width φ is the parameter that controls the number of map units around the sample's Best Matching Unit (BMU), which will be updated. For a map unit to be updated its distance from the current BMU must less than this neighbourhood width, otherwise it is not updated. The neighbourhood width φ for iteration *t* can be defined using an exponential function:

$$\varphi_t = \varphi_o \exp\left(-\frac{t\ln(\varphi_o)}{T}\right)$$

where *T* is the total number of iterations and *φ_t* is the neighbourhood width for iteration *t* using φ_o as the initial neighbourhood width. Normally, the initial neighbourhood width is defined with a value such that a large proportion of the map units will be updated in the first stage of the learning process. After that, this parameter will be decreased monotonically as the learning process continues. Therefore, it can be seen that the neighbourhood width φ_t is directly related to both the map size and the total number of the training iterations *T*.

- Learning Rate

The learning rate ψ is used to control the maximum amount a map unit can adapt its weight vector to represent a training sample. In this thesis, the similar exponential function used for the neighbourhood width is used for the learning function:

$$\psi_t = \psi_o \exp\left(-\frac{t\ln(\varphi_o)}{T}\right)$$

where *T* is the total number of iterations and ψ_t is the learning rate for iteration *t* and ψ_o is the initial learning rate. The initial learning rate ψ_o can be a number between 0 and 1. Using too high learning rate (close to 1), a larger number of learning iterations may be needed to ensure that the map has stabilised. On the other hand, using too small a value (close to 0), the weight vectors in the neighbourhood around the BMU may be inadequately trained so that regions in the map may never learn to represent one type of sample. However, to define how much each map unit around the BMU can learn is also dependent on its distance from the BMU. The units which are closer to the BMU will be adapted to be more similar to the input sample more that the unit which is further away. The overall amount a map unit can learn is proportional to both the learning rate ψ and the neighbourhood width φ. A Gaussian function can be used to describe this parameter and it is defined as the neighbourhood weight ω:

$$\omega_{kt} = \exp\left(-\frac{\mathrm{D}(\boldsymbol{m}_{\mathrm{BMU}}, \boldsymbol{m}_k)^2}{2\varphi_t^2}\right)$$

where ω_{kt} is the neighbourhood weight of map unit k for iteration t. $\mathrm{D}(\boldsymbol{m}_{\mathrm{BMU}}, \boldsymbol{m}_k)$ is the Euclidean distance between the Cartesian coordinate vector of the map unit $\boldsymbol{m}_k$ and the Cartesian coordinate vector of the BMU $\boldsymbol{m}_{\mathrm{BMU}}$.

After the neighbourhood width φ_o and the learning rate ψ_o parameters are decided, the initial map can be trained as follows:

1. A sample vector (randomly selected from the training samples for each iteration) is compared to the weight vector of each map unit using the Euclidean distance. The map unit with the most similar weight vector is declared as the 'winner' or Best Matching Unit (BMU) defined as follows:

$$s_{(x_z, w_b)} = \sqrt{\sum_{j=1}^{J} (x_{zj} - w_{bj})^2} = \min_k \{s_{(x_z, w_k)}\}$$

where b is the value of k with the most similar weight vector $\boldsymbol{w}_k$ the randomly selected sample $\boldsymbol{x}_z$:

$$b = \arg\min_k \{s_{(\boldsymbol{x}_z, \boldsymbol{w}_k)}\}$$

2. The neighbourhood width φ_t, learning rate ψ_t and neighbourhood weight ω_t are calculated for iteration t as described above.

3. The Euclidean distance between the Cartesian coordinates of each map unit $\boldsymbol{m}_k$ and the Cartesian coordinates of the BMU $\boldsymbol{m}_{\mathrm{BMU}}$ is calculated. The map units that are close enough to the BMU are declared as neighbours of the current BMU:

$$\boldsymbol{\eta}_{\mathrm{BMU}} = \{k : \mathrm{D}(\boldsymbol{m}_{\mathrm{BMU}}, \boldsymbol{m}_k) < \varphi_t\}$$

where $\boldsymbol{\eta}_{\mathrm{BMU}}$ contains the values for k for which the k^{th} map units distance from the BMU is less than the neighbourhood width in iteration φ_t.

4. The map units that have been declared as neighbours of the BMU are updated adjusting their weights proportionally to the learning rate and the neighbourhood weight for the current iteration:

$$\boldsymbol{w}_k = \begin{cases} \boldsymbol{w}_k + \omega_t \psi_t (\boldsymbol{x}_z - \boldsymbol{w}_k) & k \in \boldsymbol{\eta}_{\mathrm{BMU}} \\ \boldsymbol{w}_k & k \notin \boldsymbol{\eta}_{\mathrm{BMU}} \end{cases}$$

The BMU and its neighbouring map units are then updated to become more like the sample and all the other map units remain unchanged.

The entire process is illustrated in Figure 2.3 and is repeated until $t = T$.

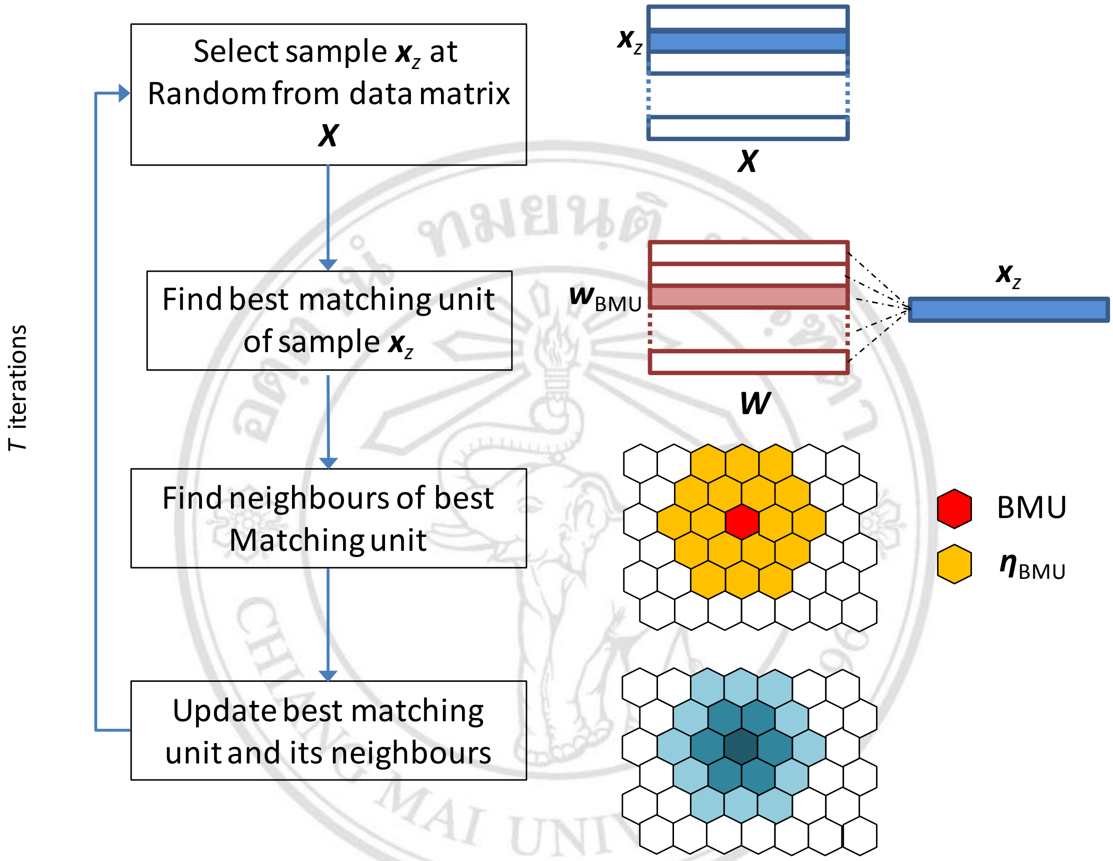
Select sample $\boldsymbol{x}_z$ at Random from data matrix $\boldsymbol{X}$
Find best matching unit of sample $\boldsymbol{x}_z$
Find neighbours of best Matching unit
Update best matching unit and its neighbours
T iterations
$\boldsymbol{x}_z$
$\boldsymbol{X}$
$\boldsymbol{w}_{\mathrm{BMU}}$
$\boldsymbol{x}_z$
$\boldsymbol{W}$
BMU
η_{BMU}

Figure 2.3 Schematic of the SOM training process

2.4.4 Supervised self-organizing map (SSOM)

Supervised self-organizing map (supervised SOM) is an extension of self-organizing map (SOM). In general, SOM focuses on representing sample data using a two-dimensional map. This map was trained based on an adaptive learning algorithm where the aim is to make the map to be well-suited for the variation of the modeling or training samples. Therefore, the structure of the samples on the original data space could be maintained as much as possible when they are placed and represented on the trained map. For a traditional SOM or an unsupervised SOM, only the information from the predictive data (***X***) is used during the training process and so the orientation of the samples on the map generated using the unsupervised learning often reflects the major variation that strongly influences over the dataset. Intentionally, for a supervised SOM, an addition of a response data is applied to each of the training samples as additional information. After that, the map was trained in the same manner of the unsupervised SOM [39]. This extension process could highlight minor variation that may be hidden by the major variation, thus enhanced the possibility of observing the clusters or groups with respect to response information provided. It is also possible to investigate whether the predictive parameters and the response behavior are alike by comparing between the component planes of each of the training parameters and the component planes of the response data or the response plane [39].

In this work, the supervised SOM was used for observing the behavior of planting conditions, some growth parameters and yield components in connecting with the response which was the contents of 2AP in the PT1 rice grains. The SOM map consists a total of 10 × 15 map units and the SOM parameters such as iteration number, initial learning rate and initial neighborhood width, were set following recommended methodology [40]. The trained map was visualized using supervised color shading and component planes, which will be discussed in detail in the Results and discussions section. The same data-preprocessing used for the PLS analysis was applied prior to the analysis of supervised SOM. It is noted here that each of the rice plants was individually used for the SOM supervised training. All of the computations in this work were conducted using in-house scripts based on Matlab software.