2 UNIT

Data Analysis

CONTENTS

PART-1

Data Analyiss : Regression Modeling, Multivarient Analysis.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.1. Write short notes on regression modeling.

Answer

1. Regression models are widely used in analytics, in general being among the most easy to understand and interpret type of analytics techniques.
2. Regression techniques allow the identification and estimation of possible relationships between a pattern or variable of interest, and factors that influence that pattern.
3. For example, a company may be interested in understanding the effectiveness of its marketing strategies.
4. A regression model can be used to understand and quantify which of its marketing activities actually drive sales, and to what extent.
5. Regression models are built to understand historical data and relationships to assess effectiveness, as in the marketing effectiveness models.
6. Regression techniques are used across a range of industries, including financial services, retail, telecom, pharmaceuticals, and medicine.

Que 2.2. What are the various types of regression analysis techniques ?

Answer

Various types of regression analysis techniques :

1. **Linear regression :** Linear regressions assumes that there is a linear relationship between the predictors (or the factors) and the target variable.
2. **Non-linear regression :** Non-linear regression allows modeling of non-linear relationships.
3. **Logistic regression :** Logistic regression is useful when our target variable is binomial (accept or reject).
4. **Time series regression :** Time series regressions is used to forecast future behavior of variables based on historical time ordered data.

Que 2.3. **Write short note on linear regression models.**

Answer

Linear regression model :

1. We consider the modelling between the dependent and one independent variable. When there is only one independent variable in the regression model, the model is generally termed as a linear regression model.
2. Consider a simple linear regression model

$$y = \beta_0 + \beta_1 X + \varepsilon$$

Where,

y is termed as the dependent or study variable and X is termed as the independent or explanatory variable.

The terms β_0 and β_1 are the parameters of the model. The parameter β_0 is termed as an intercept term, and the parameter β_1 is termed as the slope parameter.

3. These parameters are usually called as regression coefficients. The unobservable error component accounts for the failure of data to lie on the straight line and represents the difference between the true and observed realization of y.
4. There can be several reasons for such difference, such as the effect of all deleted variables in the model, variables may be qualitative, inherent randomness in the observations etc.
5. We assume that ε is observed as independent and identically distributed random variable with mean zero and constant variance σ^2 and assume that ε is normally distributed.
6. The independent variables are viewed as controlled by the experimenter, so it is considered as non-stochastic whereas y is viewed as a random variable with

$$E(y) = \beta_0 + \beta_1 X \text{ and } Var(y) = \sigma^2.$$

7. Sometimes X can also be a random variable. In such a case, instead of the sample mean and sample variance of y, we consider the conditional mean of y given $X = x$ as

$$E(y \mid x) = \beta_0 + \beta_1 x$$

and the conditional variance of y given $X = x$ as

$$Var(y \mid x) = \sigma^2$$

8. When the values of β_0, β_1, and σ^2 are known, the model is completely described. The parameters β_0, β_1 and σ^2 are generally unknown in practice and ε is unobserved. The determination of the statistical model $y = \beta_0 + \beta_1 X + \varepsilon$ depends on the determination (*i.e.* estimation) of β_0, β_1, and σ^2. In order to know the values of these parameters, n pairs of observations $(x_i, y_i)(i = 1, \ldots, n)$ on (X, y) are observed/collected and are used to determine these unknown parameters.

Que 2.4. **Write short note on multivariate analysis.**

Answer

1. Multivariate analysis (MVA) is based on the principles of multivariate statistics, which involves observation and analysis of more than one statistical outcome variable at a time.
2. These variables are nothing but prototypes of real time situations, products and services or decision making involving more than one variable.
3. MVA is used to address the situations where multiple measurements are made on each experimental unit and the relations among these measurements and their structures are important.
4. Multiple regression analysis refers to a set of techniques for studying the straight-line relationships among two or more variables.
5. Multiple regression estimates the β's in the equation

$$y_j = \beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j} + \ldots + \beta_p x_{pj} + \varepsilon_j$$

Where, the x's are the independent variables. y is the dependent variable. The subscript j represents the observation (row) number. The β's are the unknown regression coefficients. Their estimates are represented by b's. Each β represents the original unknown (population) parameter, while b is an estimate of this β. The ε_j is the error (residual) of observation j.

6. Regression problem is solved by least squares. In least squares method regression analysis, the b's are selected so as to minimize the sum of the squared residuals. This set of b's is not necessarily the set we want, since they may be distorted by outliers points that are not representative of the data. Robust regression, an alternative to least squares, seeks to reduce the influence of outliers.
7. Multiple regression analysis studies the relationship between a dependent (response) variable and p independent variables (predictors, regressors).
8. The sample multiple regression equation is

$$\hat{y}_j = b_0 + b_1 x_{2_j} + \ldots + b_p x_{p_j}$$

10. If $p = 1$, the model is called simple linear regression. The intercept, b_0, is the point at which the regression plane intersects the Y axis. The b_i are the slopes of the regression plane in the direction of x_i. These coefficients are called the partial-regression coefficients. Each partial regression coefficient represents the net effect the i^{th} variable has on the dependent variable, holding the remaining x's in the equation constant

PART-2

Bayesian Modeling, Inference and Bayesian Networks, Support Vector and Kernel Methods.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.5. Write short notes on Bayesian network.

Answer

1. Bayesian networks are a type of probabilistic graphical model that uses Bayesian inference for probability computations.
2. A Bayesian network is a directed acyclic graph in which each edge corresponds to a conditional dependency, and each node corresponds to a unique random variable.
3. Bayesian networks aim to model conditional dependence by representing edges in a directed graph.

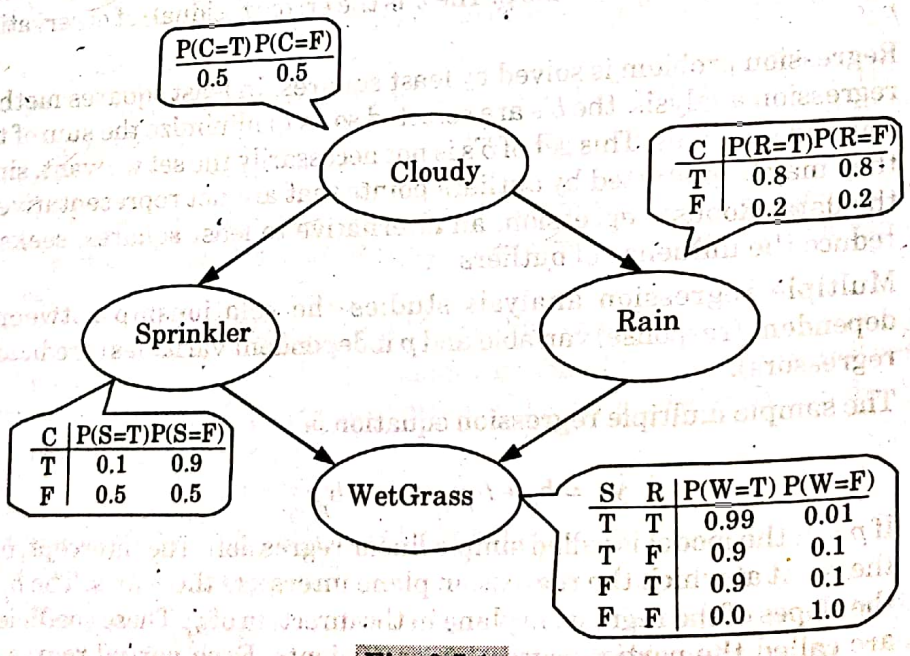

Fig. 2.5.1.

3. Through these relationships, one can efficiently conduct inference on the random variables in the graph through the use of factors.

4. Using the relationships specified by our Bayesian network, we can obtain a compact, factorized representation of the joint probability distribution by taking advantage of conditional independence.
5. Formally, if an edge (A, B) exists in the graph connecting random variables A and B, it means that $P(B \mid A)$ is a factor in the joint probability distribution, so we must know $P(B \mid A)$ for all values of B and A in order to conduct inference.
6. In the Fig. 2.5.1, since Rain has an edge going into WetGrass, it means that $P(\text{WetGrass} \mid \text{Rain})$ will be a factor, whose probability values are specified next to the WetGrass node in a conditional probability table.
7. Bayesian networks satisfy the Markov property, which states that a node is conditionally independent of its non-descendants given its parents. In the given example, this means that

   $P(\text{Sprinkler} \mid \text{Cloudy, Rain}) = P(\text{Sprinkler} \mid \text{Cloudy})$

   Since Sprinkler is conditionally independent of its non-descendant, Rain, given Cloudy.

Que 2.6. Write short notes on inference over Bayesian network.

Answer

Inference over a Bayesian network can come in two forms.

1. **First form :**
   a. The first is simply evaluating the joint probability of a particular assignment of values for each variable (or a subset) in the network.
   b. For this, we already have a factorized form of the joint distribution, so we simply evaluate that product using the provided conditional probabilities.
   c. If we only care about a subset of variables, we will need to marginalize out the ones we are not interested in.
   d. In many cases, this may result in underflow, so it is common to take the logarithm of that product, which is equivalent to adding up the individual logarithms of each term in the product.
2. **Second form :**

a. In this form, inference task is to find $P(x \mid e)$ or to find the probability of some assignment of a subset of the variables (x) given assignments of other variables (our evidence, e).

b. In the example shown in Fig. 2.6.1, we have to find

$P(\text{Sprinkler, WetGrass} \mid \text{Cloudy})$,

where {Sprinkler, WetGrass} is our x, and {Cloudy} is our e.

c. In order to calculate this, we use the fact that $P(x \mid e) = P(x, e) / P(e) = \alpha P(x, e)$, where α is a normalization constant that we will calculate at the end such that $P(x \mid e) + P(\neg x \mid e) = 1$.

d. In order to calculate $P(x, e)$, we must marginalize the joint probability distribution over the variables that do not appear in x or e, which we will denote as Y.

$$P(x \mid e) = \alpha \sum_{\forall y \in Y} P(x, e, Y)$$

e. For the given example in Fig. 2.6.1 we can calculate P(Sprinkler, WetGrass | Cloudy) as follows :

P(Sprinkler, WetGrass | Cloudy) =

$$\alpha \sum_{Rain} P(\text{WetGrass} \mid \text{Sprinkler,Rain}) P(\text{Sprinker} \mid \text{Cloudy}) P(\text{Rain} \mid \text{Cloudy})$$

P(Cloudy) =

αP(WetGrass | Sprinkler,Rain)P(Sprinker | Cloudy)P(Rain | Cloudy) P(Cloudy) +

αP(WetGrass | Sprinkler,¬Rain)P(Sprinker | Cloudy)P(¬Rain | Cloudy) P(Cloudy)

PART-3

Analysis of Time Series : Linear System Analysis of Non-Lineor Dynamics, Rule Introduction.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.7. Explain the application of time series analysis.

Answer

Applications of time series analysis :

1. **Retail sales :**
   a. For various product lines, a clothing retailer is looking to forecast future monthly sales.
   b. These forecasts need to account for the seasonal aspects of the customer's purchasing decisions.
   c. An appropriate time series model needs to account for fluctuating demand over the calendar year.
2. **Spare parts planning :**
   a. Companies service organizations have to forecast future spare part demands to ensure an adequate supply of parts to repair customer

products. Often the spares inventory consists of thousands of distinct part numbers.

b. To forecast future demand, complex models for each part number can be built using input variables such as expected part failure rates, service diagnostic effectiveness and forecasted new product shipments.

c. However, time series analysis can provide accurate short-term forecasts based simply on prior spare part demand history.

3. **Stock trading :**

a. Some high-frequency stock traders utilize a technique called pairs trading.

b. In pairs trading, an identified strong positive correlation between the prices of two stocks is used to detect a market opportunity.

c. Suppose the stock prices of Company *A* and Company *B* consistently move together.

d. Time series analysis can be applied to the difference of these companies' stock prices over time.

e. A statistically larger than expected price difference indicates that it is a good time to buy the stock of Company *A* and sell the stock of Company *B*, or vice versa.

Que 2.8. What are the components of time series ?

Answer

A time series can consist of the following components :

1. **Trends :**

a. The trend refers to the long-term movement in a time series.

b. It indicates whether the observation values are increasing or decreasing over time.

c. Examples of trends are a steady increase in sales month over month or an annual decline of fatalities due to car accidents.

2. **Seasonality :**

a. The seasonality component describes the fixed, periodic fluctuation in the observations over time.

b. It is often related to the calendar.

c. For example, monthly retail sales can fluctuate over the year due to the weather and holidays.

3. **Cyclic :**

a. A cyclic component also refers to a periodic fluctuation, which is not as fixed.

b. For example, retails sales are influenced by the general state of the economy.

Que 2.9. Explain rule induction.

Answer

1. Rule induction is a data mining process of deducing if-then rules from a dataset.
2. These symbolic decision rules explain an inherent relationship between the attributes and class labels in the dataset.
3. Many real-life experiences are based on intuitive rule induction.
4. Rule induction provides a powerful classification approach that can be easily understood by the general users.
5. It is used in predictive analytics by classification of unknown data.
6. Rule induction is also used to describe the patterns in the data.
7. The easiest way to extract rules from a data set is from a decision tree that is developed on the same data set.

Que 2.10. Explain an iterative procedure of extracting rules from data sets.

Answer

1. Sequential covering is an iterative procedure of extracting rules from the data sets.
2. The sequential covering approach attempts to find all the rules in the data set class by class.
3. One specific implementation of the sequential covering approach is called the RIPPER, which stands for Repeated Incremental Pruning to Produce Error Reduction.
4. Following are the steps in sequential covering rules generation approach :

   Step 1 : Class selection :

   a. The algorithm starts with selection of class labels one by one.

   b. The rule set is class-ordered where all the rules for a class are developed before moving on to next class.

   c. The first class is usually the least-frequent class label.

   d. From Fig. 2.10.1, the least frequent class is "+" and the algorithm focuses on generating all the rules for "+" class.

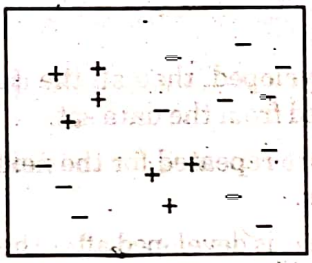

Fig. 2.10.1. Data set with two classes and two dimensions.

Step 2 : Rule development :

a. The objective in this step is to cover all "+" data points using classification rules with none or as few "–" as possible.

b. For example, in Fig. 2.10.2 , rule r_1 identifies the area of four "+" in the top left corner.

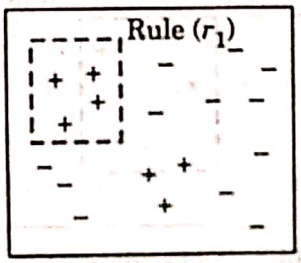

Fig. 2.10.2. Generation of ruler r_1.

c. Since this rule is based on simple logic operators in conjuncts, the boundary is rectilinear.

d. Once rule r_1 is formed, the entire data points covered by r_1 are eliminated and the next best rule is found from data sets.

Step 3 : Learn-One-Rule :

a. Each rule r_1 is grown by the learn-one-rule approach.

b. Each rule starts with an empty rule set and conjuncts are added one by one to increase the rule accuracy.

c. Rule accuracy is the ratio of amount of "+" covered by the rule to all records covered by the rule :

$$\text{Rule accuracy } A(r_i) = \frac{\text{Correct records by rule}}{\text{All records covered by the rule}}$$

d. Learn-one-rule starts with an empty rule set: if {} then class = "+".

e. The accuracy of this rule is the same as the proportion of + data points in the data set. Then the algorithm greedily adds conjuncts until the accuracy reaches 100 %.

f. If the addition of a conjunct decreases the accuracy, then the algorithm looks for other conjuncts or stops and starts the iteration of the next rule.

Step 4 : Next rule :

a. After a rule is developed, then all the data points covered by the rule are eliminated from the data set.

b. The above steps are repeated for the next rule to cover the rest of the "+" data points.

c. In Fig. 2.10.3, rule r_2 is developed after the data points covered by r_1 are eliminated.

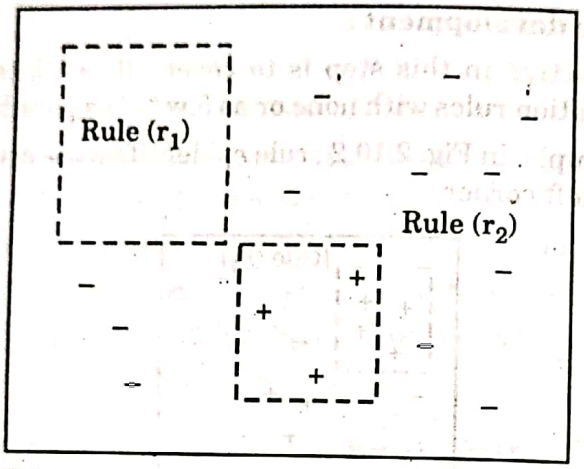

Fig. 2.10.3. Elimination of r1 data points and next rule.

Step 5 : Development of rule set :

a. After the rule set is developed to identify all "+" data points, the rule model is evaluated with a data set used for pruning to reduce generalization errors.

b. The metric used to evaluate the need for pruning is $(p - n)/(p + n)$, where p is the number of positive records covered by the rule and n is the number of negative records covered by the rule.

c. All rules to identify "+" data points are aggregated to form a rule group.

PART-4

Neural Networks : Learning and Generalization, Competitive Learning, Principal Component Analysis and Neural Networks.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.11. Describe supervised learning and unsupervised learning.

Answer

Supervised learning :

1. Supervised learning is also known as associative learning, in which the network is trained by providing it with input and matching output patterns.
2. Supervised training requires the pairing of each input vector with a target vector representing the desired output.
3. The input vector together with the corresponding target vector is called training pair.
4. To solve a problem of supervised learning following steps are considered :
   a. Determine the type of training examples.
   b. Gathering of a training set.
   c. Determine the input feature representation of the learned function.
   d. Determine the structure of the learned function and corresponding learning algorithm.
   e. Complete the design.
5. Supervised learning can be classified into two categories :
   i. Classification ii. Regression

Unsupervised learning :

1. Unsupervised learning, an output unit is trained to respond to clusters of pattern within the input.
3. In this method of training, the input vectors of similar type are grouped without the use of training data to specify how a typical member of each group looks or to which group a member belongs.
3. Unsupervised training does not require a teacher; it requires certain guidelines to form groups.
4. Unsupervised learning can be classified into two categories :
   i. Clustering ii. Association

Que 2.12. Differentiate between supervised learning and unsupervised learning.

Answer

Difference between supervised and unsupervised learning :

S. No.	Supervised learning	Unsupervised learning
1.	It uses known and labeled data as input.	It uses unknown data as input.
2.	Computational complexity is very complex.	Computational complexity is less.
3.	It uses offline analysis.	It uses real time analysis of data.
4.	Number of classes is known.	Number of classes is not known.
5.	Accurate and reliable results.	Moderate accurate and reliable results.

Que 2.13. What is the multilayer perceptron model ? Explain it.

Answer

1. Multilayer perceptron is a class of feed forward artificial neural network.
2. Multilayer perceptron model has three layers; an input layer, and output layer, and a layer in between not connected directly to the input or the output and hence, called the hidden layer.
3. For the perceptrons in the input layer, we use linear transfer function, and for the perceptrons in the hidden layer and the output layer, we use sigmoidal or squashed-*S* function.
4. The input layer serves to distribute the values they receive to the next layer and so, does not perform a weighted sum or threshold.
5. The input-output mapping of multilayer perceptron is shown in Fig. 2.13.1 and is represented by

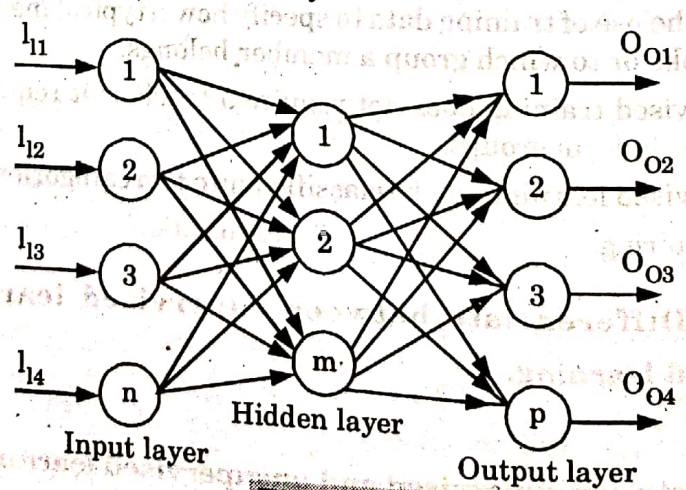

Fig. 2.13.1.

6. Multilayer perceptron does not increase computational power over a single layer neural network unless there is a non-linear activation function between layers.

Que 2.14. Draw and explain the multiple perceptron with its learning algorithm.

Answer

1. The perceptrons which are arranged in layers are called multilayer (multiple) perceptron.
2. This model has three layers : an input layer, output layer and one or more hidden layer.
3. For the perceptrons in the input layer, the linear transfer function used and for the perceptron in the hidden layer and output layer, the sigmoidal or squashed-S function is used. The input signal propagates through the network in a forward direction.
4. In the multilayer perceptron bias $b(n)$ is treated as a synaptic weight driven by fixed input equal to + 1.

$$x(n) = [+1, x_1(n), x_2(n), \ldots\ldots\ldots x_m(n)]^T$$

where n denotes the iteration step in applying the algorithm.

5. Correspondingly we define the weight vector as :

$$w(n) = [b(n), w_1(n), w_2(n)\ldots\ldots\ldots, w_m(n)]^T$$

6. Accordingly the linear combiner output is written in the compact form

$$V(n) = \sum_{i=0}^{m} w_i(n)x_i(n) = w^T(n)\, x(n)$$

Architecture of multilayer perceptron :

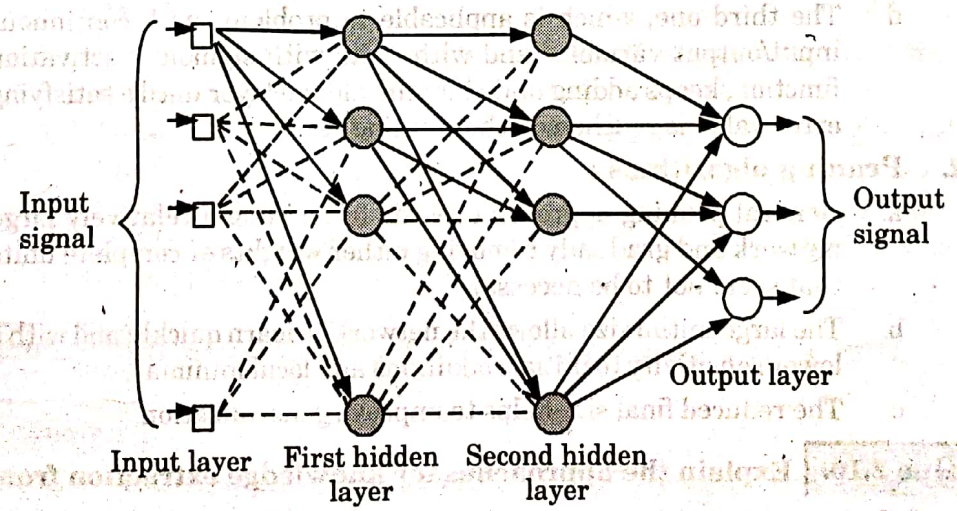

Fig. 2.14.1.

7. Fig. 2.14.1 shows the architectural model of multilayer perceptron with two hidden layer and an output layer.
8. Signal flow through the network progresses in a forward direction, from the left to right and on a layer-by-layer basis.

Learning algorithm :

1. If the nth number of input set $x(n)$, is correctly classified into linearly separable classes, by the weight vector $w(n)$ then no adjustment of weights are done.

   $$w(n+1) = w(n)$$

   If $w^T x(n) > 0$ and $x(n)$ belongs to class G_1.

   $$w(n+1) = w(n)$$

   If $w^T x(n) \leq 0$ and $x(n)$ belongs to class G_2.
2. Otherwise, the weight vector of the perceptron is updated in accordance with the rule.

Que 2.15. Explain the algorithm to optimize the network size.

Answer

Algorithms to optimize the network size are :

1. **Growing algorithms :**
   a. This group of algorithms begins with training a relatively small neural architecture and allows new units and connections to be added during the training process, when necessary.
   b. Three growing algorithms are commonly applied: the upstart algorithm, the tiling algorithm, and the cascade correlation.
   c. The first two apply to binary input/output variables and networks with step activation function.
   d. The third one, which is applicable to problems with continuous input/output variables and with units with sigmoidal activation function, keeps adding units into the hidden layer until a satisfying error value is reached on the training set.
2. **Pruning algorithms :**
   a. General pruning approach consists of training a relatively large network and gradually removing either weights or complete units that seem not to be necessary.
   b. The large initial size allows the network to learn quickly and with a lower sensitivity to initial conditions and local minima.
   c. The reduced final size helps to improve generalization.

Que 2.16. Explain the approaches for knowledge extraction from multilayer perceptrons.

Answer

Approach for knowledge extraction from multilayer perceptrons :

a. Global approach :

1. This approach extracts a set of rules characterizing the behaviour of the whole network in terms of input/output mapping.
2. A tree of candidate rules is defined. The node at the top of the tree represents the most general rule and the nodes at the bottom of the tree represent the most specific rules.
3. Each candidate symbolic rule is tested against the network's behaviour, to see whether such a rule can apply.
4. The process of rule verification continues until most of the training set is covered.
5. One of the problems connected with this approach is that the number of candidate rules can become huge when the rule space becomes more detailed.

b. Local approach :

1. This approach decomposes the original multilayer network into a collection of smaller, usually single-layered, sub-networks, whose input/output mapping might be easier to model in terms of symbolic rules.
2. Based on the assumption that hidden and output units, though sigmoidal, can be approximated by threshold functions, individual units inside each sub-network are modeled by interpreting the incoming weights as the antecedent of a symbolic rule.
3. The resulting symbolic rules are gradually combined together to define a more general set of rules that describes the network as a whole.
4. The monotonicity of the activation function is required, to limit the number of candidate symbolic rules for each unit.
5. Local rule-extraction methods usually employ a special error function and/or a modified learning algorithm, to encourage hidden and output units to stay in a range consistent with possible rules and to achieve networks with the smallest number of units and weights.

Que 2.17. Discuss the selection of various parameters in BPN.

Answer

Selection of various parameters in BPN (Back Propagation Network) :

1. Number of hidden nodes :

i. The guiding criterion is to select the minimum nodes which would not impair the network performance so that the memory demand for storing the weights can be kept minimum.

ii. When the number of hidden nodes is equal to the number of training patterns, the learning could be fastest.

iii. In such cases, Back Propagation Network (BPN) remembers training patterns losing all generalization capabilities.

iv. Hence, as far as generalization is concerned, the number of hidden nodes should be small compared to the number of training patterns (say 10:1).

2. Momentum coefficient (α) :

i. The another method of reducing the training time is the use of momentum factor because it enhances the training process.

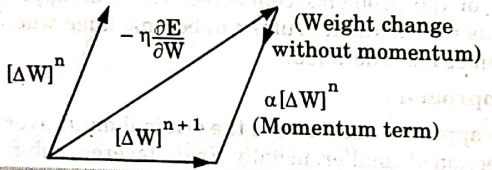

Fig. 2.17.1. Influence of momentum term on weight change.

ii. The momentum also overcomes the effect of local minima.

iii. It will carry a weight change process through one or local minima and get it into global minima.

3. Sigmoidal gain (λ) :

i. When the weights become large and force the neuron to operate in a region where sigmoidal function is very flat, a better method of coping with network paralysis is to adjust the sigmoidal gain.

ii. By decreasing this scaling factor, we effectively spread out sigmoidal function on wide range so that training proceeds faster.

4. Local minima :

i. One of the most practical solutions involves the introduction of a shock which changes all weights by specific or random amounts.

ii. If this fails, then the solution is to re-randomize the weights and start the training all over.

iii. Simulated annealing used to continue training until local minima is reached.

iv. After this, simulated annealing is stopped and BPN continues until global minimum is reached.

v. In most of the cases, only a few simulated annealing cycles of this two-stage process are needed.

Answer

Approach for knowledge extraction from multilayer perceptrons :

a. Global approach :

1. This approach extracts a set of rules characterizing the behaviour of the whole network in terms of input/output mapping.
2. A tree of candidate rules is defined. The node at the top of the tree represents the most general rule and the nodes at the bottom of the tree represent the most specific rules.
3. Each candidate symbolic rule is tested against the network's behaviour, to see whether such a rule can apply.
4. The process of rule verification continues until most of the training set is covered.
5. One of the problems connected with this approach is that the number of candidate rules can become huge when the rule space becomes more detailed.

b. Local approach :

1. This approach decomposes the original multilayer network into a collection of smaller, usually single-layered, sub-networks, whose input/output mapping might be easier to model in terms of symbolic rules.
2. Based on the assumption that hidden and output units, though sigmoidal, can be approximated by threshold functions, individual units inside each sub-network are modeled by interpreting the incoming weights as the antecedent of a symbolic rule.
3. The resulting symbolic rules are gradually combined together to define a more general set of rules that describes the network as a whole.
4. The monotonicity of the activation function is required, to limit the number of candidate symbolic rules for each unit.
5. Local rule-extraction methods usually employ a special error function and/or a modified learning algorithm, to encourage hidden and output units to stay in a range consistent with possible rules and to achieve networks with the smallest number of units and weights.

Que 2.17. Discuss the selection of various parameters in BPN.

Answer

Selection of various parameters in BPN (Back Propagation Network) :

1. Number of hidden nodes :

i. The guiding criterion is to select the minimum nodes which would not impair the network performance so that the memory demand for storing the weights can be kept minimum.

ii. When the number of hidden nodes is equal to the number of training patterns, the learning could be fastest.

iii. In such cases, Back Propagation Network (BPN) remembers training patterns losing all generalization capabilities.

iv. Hence, as far as generalization is concerned, the number of hidden nodes should be small compared to the number of training patterns (say 10:1).

2. Momentum coefficient (α) :

i. The another method of reducing the training time is the use of momentum factor because it enhances the training process.

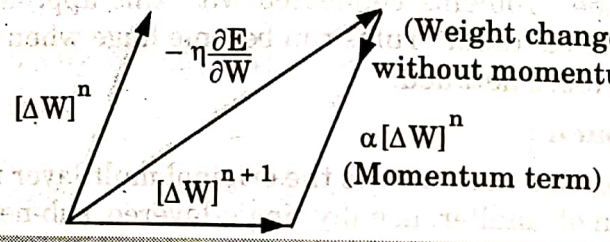

Fig. 2.17.1. Influence of momentum term on weight change.

ii. The momentum also overcomes the effect of local minima.

iii. It will carry a weight change process through one or local minima and get it into global minima.

3. Sigmoidal gain (λ) :

i. When the weights become large and force the neuron to operate in a region where sigmoidal function is very flat, a better method of coping with network paralysis is to adjust the sigmoidal gain.

ii. By decreasing this scaling factor, we effectively spread out sigmoidal function on wide range so that training proceeds faster.

4. Local minima :

i. One of the most practical solutions involves the introduction of a shock which changes all weights by specific or random amounts.

ii. If this fails, then the solution is to re-randomize the weights and start the training all over.

iii. Simulated annealing used to continue training until local minima is reached.

iv. After this, simulated annealing is stopped and BPN continues until global minimum is reached.

v. In most of the cases, only a few simulated annealing cycles of this two-stage process are needed.

5. Learning coefficient (η) :

i. The learning coefficient cannot be negative because this would cause the change of weight vector to move away from ideal weight vector position.

ii. If the learning coefficient is zero, no learning takes place and hence, the learning coefficient must be positive.

iii. If the learning coefficient is greater than 1, the weight vector will overshoot from its ideal position and oscillate.

iv. Hence, the learning coefficient must be between zero and one.

Que 2.18. What is learning rate ? What is its function ?

Answer

1. Learning rate is a constant used in learning algorithm that define the speed and extend in weight matrix corrections.
2. Setting a high learning rate tends to bring instability and the system is difficult to converge even to a near optimum solution.
3. A low value will improve stability, but will slow down convergence.

Learning function :

1. In most applications the learning rate is a simple function of time for example L.R. = $1/(1 + t)$.
2. These functions have the advantage of having high values during the first epochs, making large corrections to the weight matrix and smaller values later, when the corrections need to be more precise.
3. Using a fuzzy controller to adaptively tune the learning rate has the added advantage of bringing all expert knowledge in use.
4. If it was possible to manually adapt the learning rate in every epoch, we would surely follow rules of the kind listed below :
   a. If the change in error is small, then increase the learning rate.
   b. If there are a lot of sign changes in error, then largely decrease the learning rate.
   c. If the change in error is small and the speed of error change is small, then make a large increase in the learning rate.

Que 2.19. Explain competitive learning.

Answer

1. Competitive learning is a form of unsupervised learning in artificial neural networks, in which nodes compete for the right to respond to a subset of the input data.
2. A variant of Hebbian learning, competitive learning works by increasing the specialization of each node in the network. It is well suited to finding clusters within data.

3. Models and algorithms based on the principle of competitive learning include vector quantization and self-organizing maps.
4. In a competitive learning model, there are hierarchical sets of units in the network with inhibitory and excitatory connections.
5. The excitatory connections are between individual layers and the inhibitory connections are between units in layered clusters.
6. Units in a cluster are either active or inactive.
7. There are three basic elements to a competitive learning rule :
   a. A set of neurons that are all the same except for some randomly distributed synaptic weights, and which therefore respond differently to a given set of input patterns.
   b. A limit imposed on the "strength" of each neuron.
   c. A mechanism that permits the neurons to compete for the right to respond to a given subset of inputs, such that only one output neuron (or only one neuron per group), is active (*i.e.*, "on") at a time. The neuron that wins the competition is called a "winner-take-all" neuron.

Que 2.20. Explain Principle Component Analysis (PCA) in data analysis.

Answer

1. PCA is a method used to reduce number of variables in dataset by extracting important one from a large dataset.
2. It reduces the dimension of our data with the aim of retaining as much information as possible.
3. In other words, this method combines highly correlated variables together to form a smaller number of an artificial set of variables which is called principal components (PC) that account for most variance in the data.
4. A principal component can be defined as a linear combination of optimally-weighted observed variables.
5. The first principal component retains maximum variation that was present in the original components.
6. The principal components are the eigenvectors of a covariance matrix, and hence they are orthogonal.
7. The output of PCA are these principal components, the number of which is less than or equal to the number of original variables.
8. The PCs possess some useful properties which are listed below :
   a. The PCs are essentially the linear combinations of the original variables and the weights vector.
   b. The PCs are orthogonal.

c. The variation present in the PC decrease as we move from the 1st PC to the last one.

PART-5

Fuzzy Logic : Extracting Fuzzy Models From Data, Fuzzy Decision Trees, Stochastic Search Methods.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 2.21. Define fuzzy logic and its importance in our daily life. What is role of crisp sets in fuzzy logic ?

Answer

1. Fuzzy logic is an approach to computing based on "degrees of truth" rather than "true or false" (1 or 0).
2. Fuzzy logic includes 0 and 1 as extreme cases of truth but also includes the various states of truth in between.
3. Fuzzy logic allows inclusion of human assessments in computing problems.
4. It provides an effective means for conflict resolution of multiple criteria and better assessment of options.

Importance of fuzzy logic in daily life :

1. Fuzzy logic is essential for the development of human-like capabilities for AI.
2. It is used in the development of intelligent systems for decision making, identification, optimization, and control.
3. Fuzzy logic is extremely useful for many people involved in research and development including engineers, mathematicians, computer software developers and researchers.
4. Fuzzy logic has been used in numerous applications such as facial pattern recognition, air conditioners, vacuum cleaners, weather forecasting systems, medical diagnosis and stock trading.

Role of crisp sets in fuzzy logic :

1. It contains the precise location of the set boundaries.
2. It provides the membership value of the set.

Que 2.22. Define classical set and fuzzy sets. State the importance of fuzzy sets.

Answer

Classical set :

1. Classical set is a collection of distinct objects.
2. Each individual entity in a set is called a member or an element of the set.
3. The classical set is defined in such a way that the universe of discourse is splitted into two groups as members and non-members.

Fuzzy set :

1. Fuzzy set is a set having degree of membership between 1 and 0.
2. Fuzzy sets $\tilde{A}$ in the universe of discourse U can be defined as set of ordered pair and it is given by

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x) \mid x \in U)\}$$

Where $\mu_{\tilde{A}}$ is the degree of membership of x in $\tilde{A}$.

Importance of fuzzy set :

1. It is used for the modeling and inclusion of contradiction in a knowledge base.
2. It also increases the system autonomy.
3. It acts as an important part of microchip processor-based appliances.

Que 2.23. Compare and contrast classical logic and fuzzy logic.

Answer

S.No.	Crisp (classical) logic	Fuzzy logic
1.	In classical logic an element either belongs to or does not belong to a set.	Fuzzy logic supports a flexible sense of membership of elements to a set.
2.	Crisp logic is built on a 2-state truth values (True/False).	Fuzzy logic is built on a multistate truth values.
3.	The statement which is either 'True' or 'False' but not both is called a proposition in crisp logic.	A fuzzy proposition is a statement which acquires a fuzzy truth value.
4.	Law of excluded middle and law of non-contradiction holds good in crisp logic.	Law of excluded middle and law of contradiction are violated.

Que 2.24. Define the membership function and state its importance in fuzzy logic. Also discuss the features of membership functions.

Answer

Membership function :

1. A membership function for a fuzzy set A on the universe of discourse X is defined as $\mu_A : X \rightarrow [0,1]$, where each element of X is mapped to a value between 0 and 1.
2. This value, called membership value or degree of membership, quantifies the grade of membership of the element in X to the fuzzy set A.
3. Membership functions characterize fuzziness (*i.e.*, all the information in fuzzy set), whether the elements in fuzzy sets are discrete or continuous.
4. Membership functions can be defined as a technique to solve practical problems by experience rather than knowledge.
5. Membership functions are represented by graphical forms.

Importance of membership function in fuzzy logic :

1. It allows us to graphically represent a fuzzy set.
2. It helps in finding different fuzzy set operation.

Features of membership function :

1. **Core :**
   a. The core of a membership function for some fuzzy set $\tilde{A}$ is defined as that region of the universe that is characterized by complete and full membership in the set.
   b. The core comprises those elements x of the universe such that $\mu_{\tilde{A}}(x) = 1$.
2. **Support :**
   a. The support of a membership function for some fuzzy set $\tilde{A}$ is defined as that region of the universe that is characterized by nonzero membership in the set $\tilde{A}$.
   b. The support comprises those elements x of the universe such that $\mu_{\tilde{A}}(x) > 0$.

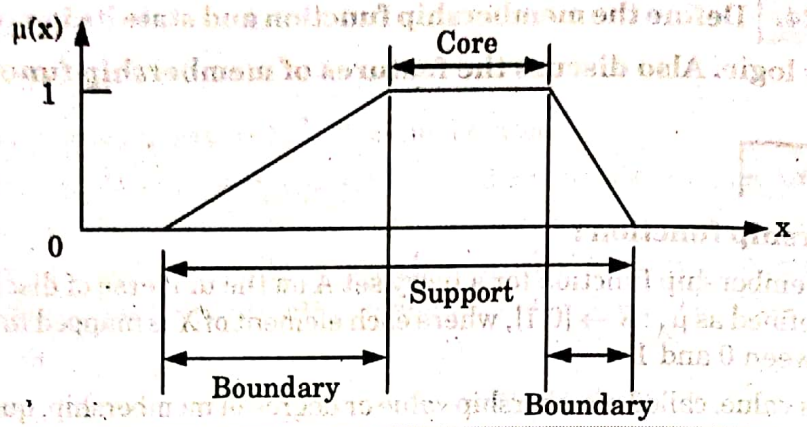

Fig. 2.24.1. Core, support, and boundaries of a fuzzy set.

3. **Boundaries :**

   a. The boundaries of a membership function for some fuzzy set $\tilde{A}$ are defined as that region of the universe containing elements that have a non-zero membership but not complete membership.

   b. The boundaries comprise those elements x of the universe such that $0 < \mu_{\tilde{A}}(x) < 1$.

Que 2.25. **Explain the inference in fuzzy logic.**

Answer

Fuzzy Inference :

1. Inferences is a technique where facts, premises $F_1, F_2, \ldots\ldots, F_n$ and a goal G is to be derived from a given set.
2. Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic.
3. The mapping then provides a basis from which decisions can be made.
4. Fuzzy inference (approximate reasoning) refers to computational procedures used for evaluating linguistic (IF-THEN) descriptions.
5. **The two important inferring procedures are :**

i. **Generalized Modus Ponens (GMP) :**

1. GMP is formally stated as

   If x is $\tilde{A}$ THEN y is $\tilde{B}$

   $$\frac{x \text{ is } \tilde{A}'}{y \text{ is } \tilde{B}'}$$

   Here, $\tilde{A}$, $\tilde{B}$, $\tilde{A}'$ and $\tilde{B}'$ are fuzzy terms.
2. Every fuzzy linguistic statement above the line is analytically known and what is below is analytically unknown.

Here $\tilde{B}' = \tilde{A}' \circ \tilde{R}(x, y)$

where 'o' denotes max-min composition (IF-THEN relation)

3. The membership function is

$$\mu_{\tilde{B}'}(y) = \max(\min(\mu_{\tilde{A}'}(x), \mu_{\tilde{R}}(x, y)))$$

where $\mu_{\tilde{B}'}(y)$ is membership function of $\tilde{B}'$, $\mu_{\tilde{A}'}(x)$ is membership function of $\tilde{A}'$ and $\mu_{\tilde{R}}(x, y)$ is the membership function of implication relation.

ii. Generalized Modus Tollens (GMT)

1. GMT is defined as

If x is $\tilde{A}$. Then y is $\tilde{B}$

y is $\tilde{B}'$

---

x is $\tilde{A}'$

2. The membership of $\tilde{A}'$ is computed as

$$\tilde{A}' = \tilde{B}' \circ \tilde{R}(x, y)$$

3. In terms of membership function

$$\mu_{\tilde{A}'}(x) = \max(\min(\mu_{\tilde{B}'}(y), \mu_{\tilde{R}}(x, y)))$$

Que 2.26. Explain Fuzzy Decision Tree (FDT).

Answer

1. Decision trees are one of the most popular methods for learning and reasoning from instances.
2. Given a set of n input-output training patterns $D = \{(X^i, y^i)\, i = 1, ..., n\}$, where each training pattern X^i has been described by a set of p conditional (or input) attributes $(x_1, ..., x_p)$ and one corresponding discrete class label y^i where $y^i \in \{1, ..., q\}$ and q is the number of classes.
3. The decision attribute y^i represents a posterior knowledge regarding the class of each pattern.
4. An arbitrary class has been indexed by l $(1 \le l \le q)$ and each class l has been modeled as a crisp set.
5. The membership degree of the i^{th} value of the decision attribute y^i concerning the i^{th} class is defined as follows :

$$\mu_l(y^i) = \begin{cases} 1, & \text{if } y^i \text{ belong to } l^{th} \text{ class;} \\ 0, & \text{otherwise.} \end{cases}$$

6. The architecture of induction of FDT is given in Fig. 2.26.1

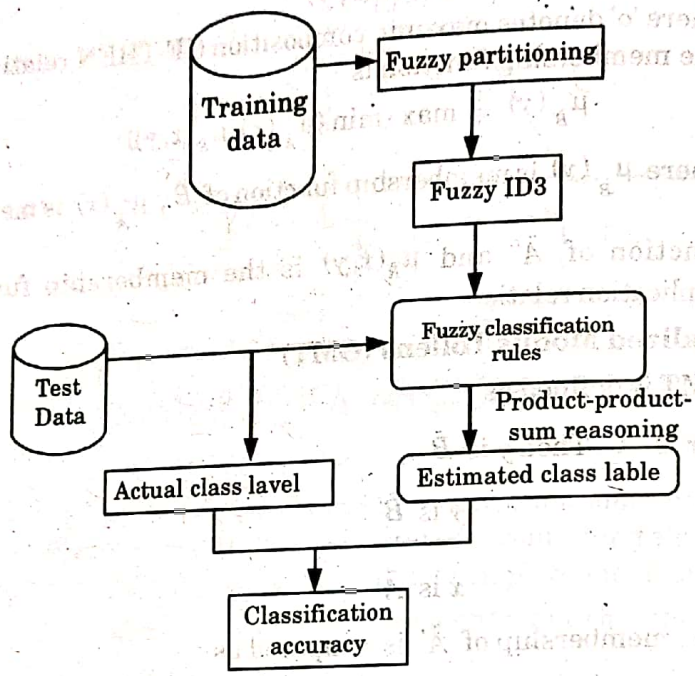

Fig. 2.26.1. Architecture of Fuzzy decision tree induction.

7. The generation of FDT for pattern classification consists of three major steps namely fuzzy partitioning (clustering), induction of FDT and fuzzy rule inference for classification.
8. The first crucial step in the induction process of FDT is the fuzzy partitioning of input space using any fuzzy clustering techniques.
9. FDTs are constructed using any standard algorithm like Fuzzy ID3 where we follow a top-down, recursive divide and conquer approach, which makes locally optimal decisions at each node.
10. As the tree is being built, the training set is recursively partitioned into smaller subsets and the generated fuzzy rules are used to predict the class of an unseen pattern by applying suitable fuzzy inference/reasoning mechanism on the FDT.
11. The general procedure for generating fuzzy decision trees using Fuzzy ID3 is as follows :

Prerequisites : A Fuzzy partition space, leaf selection threshold β_{th} and the best node selection criterion

Procedure :

While there exist candidate nodes

DO Select one of them using a search strategy.

Generate its child-nodes according to an expanded attribute obtained by the given heuristic.

Check child nodes for the leaf selection threshold.

Child-nodes meeting the leaf threshold have to be terminated as leaf-nodes.

The remaining child-nodes are regarded as new candidate node.

end

Que 2.27. Write short notes on extracting grid-based fuzzy models from data.

Answer

1. Grid-based rule sets model each input variable through a usually small set of linguistic values.
2. The resulting rule base uses all or a subset of all possible combinations of these linguistic values for each variable, resulting in a global granulation of the feature space into "tiles":

$R_{1,\ldots,1}$: IF x_1 IS $A_{1,1}$ AND ... AND x_n IS $A_{1,n}$ THEN ...

$\vdots \qquad \vdots$

$R_{1,\ldots,ln}$: IF x_1 IS $A_{1,1}$ AND ... AND x_n IS $A_{ln,n}$ THEN ...

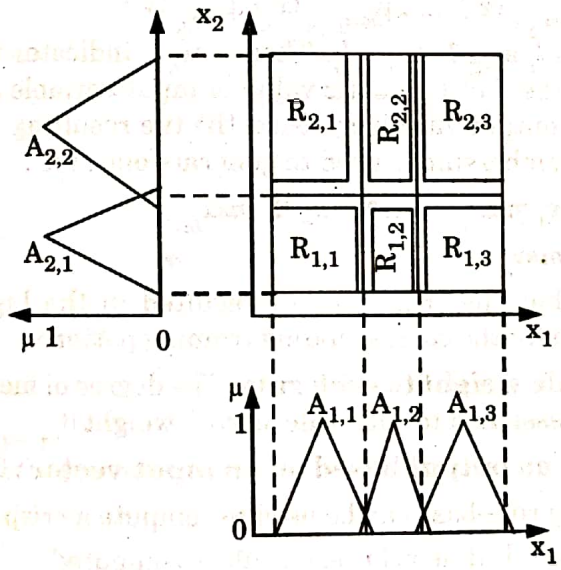

Fig. 2.27.1. A global granulation of the input space using three membership functions for x_1 and two for x_2.

where l_1 $(1 < i < n)$ indicates the numbers of linguistic values for variable i in the n-dimensional feature space. Fig. 2.27.1 illustrates this approach in two dimensions with $l_1 = 3$ and $l_2 = 2$.

3. Extracting grid-based fuzzy models from data is straightforward when the input granulation is fixed, that is, the antecedents of all rules are

predefined. Then only a matching consequent for each rule needs to be found.

4. After predefinition of the granulation of all input variables and also the output variable, one sweep through the entire dataset determines the closest example to the geometrical center of each rule, assigning the closest output fuzzy value to the corresponding rule:

1. **Granulate the input and output space :**
   a. Divide each variable X_i into l_i equidistant triangular membership functions.
   b. Similarly the granulation into l_y membership functions for the output variable y is determined, resulting in the typical overlapping distribution of triangular membership functions.
   c. Fig. 2.27.1 illustrates this approach in two dimensions with respect to membership functions, resulting in six tiles.

2. **Generate fuzzy rules from given data :**
   a. For the example in Fig. 2.27.1, this means that we have to determine the best consequence for each rule.
   b. For each example pattern (x, y) the degree of membership to each of the possible tiles is determined :

$$\min\{\mu_{msx_{j_1,1}}(x_1), \ldots, \mu_{msx_{j_n,n}}(x_n), \mu_{msy_{j_y}}(y)\}$$

with $1 \le j_i \le l_i$ and $1 < j_y < l_y$. Then $msx_{j_i,i}$, indicates the membership function of the j_i-th linguistic value of input variable i and similar for msy for the output variable y. Next the tile resulting in the maximum degree of membership is used to generate one rule :

$R_{(j_1,\ldots j_n)}$: IF x_1 $msx_{j_1,1}$ ... AND x_n IS $msx_{j_n,n}$
THEN y IS msy_{j_y}

assuming that tile $(j_1, \ldots j_n, j_y)$ resulted in the highest degree of membership for the corresponding training pattern.

3. **Assign a rule weight to each rule :** The degree of membership will in addition be assigned to each rule as rule-weight $\beta_{(j_1,\ldots j_n)}$.

4. **Determine an output based on an input-vector :** Given an input x the resulting rule-base can be used to compute a crisp output $\hat{y}$. First the degree of fulfillment for each rule is computed :

$$\mu(j_1,\ldots,j_n)(x) = \min\{\mu_{msx_{j_1,1}}(x_1), \ldots, \mu_{msx_{j_n,n}}(x_n)\}$$

then the output $\hat{y}$ is combined through a centroid defuzzification formula :

$$\hat{y} = \frac{\sum_{j_1=1,\ldots m\, j_n=1}^{l_1,\ldots,l_n} \beta_{(j_1,\ldots j_n)} \cdot \mu_{(j_1,\ldots,j_n)}(x) \cdot \overline{y}_{(j_1,\ldots,j_n)}}{\sum_{j_1=1,\ldots m\, j_n=1}^{l_1,\ldots,l_n} \beta_{(j_1,\ldots j_n)} \cdot \mu_{(j_1,\ldots,j_n)}(x)}$$

where $\bar{y}_{(j_1,\ldots,j_n)}$ denotes the center of the output region of the corresponding rule with index $(j_1, \ldots j_n)$.