MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L - 1

LECTURE HANDOUTS

CSE **IV/VII - B**

Course Name with Code :Machine Learning Techniques -16CSE14

Course Teacher : DrN.Naveenkumar

Unit : I - Introduction and Supervised Learning Date of Lecture:

Topic of Lecture: Introduction to Machine Learning- basic concepts in machine learning

Introduction: (Maximum 5 sentences) : Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Theory of computation basics
Non-Deterministic Finite Automata
Deterministic Finite Automata

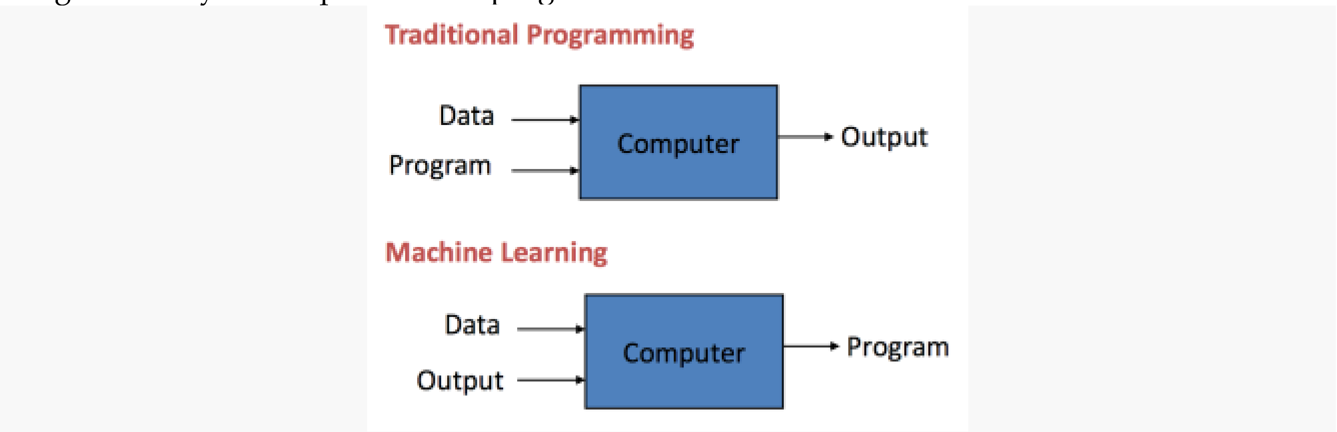
Detailed content of the Lecture:

- Machine learning provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

Machine learning is the way to make programming scalable.**Traditional Programming**: Data and program is run on the computer to produce the output.

Machine Learning: Data and output is run on the computer to create a program. This program can be used in traditional programming.

Machine learning is like farming or gardening. Seeds is the algorithms, nutrients is the data, the gardner is you and plants is the programs.



Traditional Programming vs Machine Learning

- If we could understand how to program them to learn to improve automatically with experience, the impact would be dramatic.
- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- We have a model defined up to some parameters, and learning is the execution of a computer program to optimize the parameters of the model using the training data or past experience.
- The model may be *predictive* to make predictions in the future, or *descriptive* to gain knowledge from data, or both.
- Machine learning uses the theory of statistics in building mathematical models, because the core task is making inference from a sample.
- The role of computer science is twofold: First, in training, we need efficient algorithms to solve the optimization problem, as well as to store and process the massive amount of data we generally have.
- Second, once a model is learned, its representation and algorithmic solution for inference needs to be efficient as well.
- In certain applications, the efficiency of the learning or inference algorithm, namely, its space and time complexity, may be as important as its predictive accuracy.
- There are the eyes, the nose, and the mouth, located in certain places on the face. Each person's face is a pattern composed of a particular combination of these.
- By analyzing sample face images of a person, a learning program captures the pattern specific to that person and then recognizes by checking for this pattern in a given image. This is one example of *pattern recognition*.
- We may not be able to identify the process completely, but we believe we can construct *a* good and useful approximation.
- Application of machine learning methods to large databases is called data mining.
- Machine learning also helps us find solutions to many problems in vision, speech recognition, and robotics.

Video Content / Details of website for further learning (if any):
https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop
https://www.youtube.com/watch?v=IpGxLWOIZy4

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 1-4

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

DESIGNING YOUR FUTURE
Estd. 2000

IQAC

L - 2

LECTURE HANDOUTS

CSE **IV/VII - B**

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning **Date of Lecture:**

Topic of Lecture: Examples of machine learning applications

Introduction: (Maximum 5 sentences) Machine learning concept is applied to many of the applications like associations,classification,Regression,etc.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Supervised learning basics
Unsupervised learning basics
Reinforcement Learning basics

Detailed content of the Lecture:

Examples of Machine Learning Applications

*** Learning Associations**

- In the case of retail—for example, a supermarket chain—one application of machine learning is *basket analysis*, which is finding associations between products bought by customers.
- If people who buy *X* typically also buy *Y*, and if there is a customer who buys *X* and does not buy *Y*, he or she is a potential *Y* customer. Once we find such customers, we can target them for cross-selling.

Association Rule:

- In finding an *association rule*, association rule we are interested in learning a conditional probability of the form $P(Y|X)$ where *Y* is the product we would like to condition on *X*, which is the product or the set of products which we know that the customer has already purchased.

* **Supervised Learning:**(also called inductive learning) Training data includes desired outputs. This is spam this is not, learning is supervised.

*** Classification**

- This classification is an example of a *classification* problem where there are two classes:

low-risk and high-risk customers. The information about a customer makes up the *input* to the classifier whose task is to assign the input to one of the two classes.

Classification Rule:

IF income> $\theta 1$ AND savings> $\theta 2$ THEN low-risk ELSE high-risk

Discriminant:

- For suitable values of $\theta 1$ and $\theta 2$ (see figure 1.1). This is an example of a *discriminant*; it is function that separates the examples of different classes.

Knowledge Extraction:

- Learning a rule from data also allows *knowledge extraction*. The rule is a simple model that explains the data, and looking at this model we have an explanation about the process underlying the data.

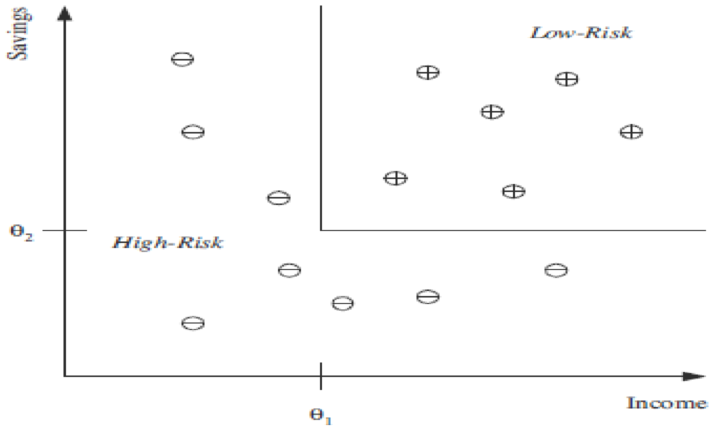


Figure 1.1 Example for classification

*** Regression**

- Let *X* denote the car attributes and *Y* be the price of the car. Again surveying the past transactions, we can collect a training data and the machine learning program fits a function to this data to learn *Y* as a function of *X*. An example is given in figure 1.2 where the fitted function is of the form

$$y = wx + w0$$

For suitable values of *w* and *w*0.

*** Unsupervised Learning**

- Training data does not include desired outputs. The aim is to find the regularities in the input. There is a structure to the input space such that certain patterns occur more often than others, and we want to see what generally happens and what does not.

*** Reinforcement Learning**

- In such a case, the machine learning program should be able to assess the goodness of policies and learn from past good action sequences to be able to generate a policy. Such learning methods are called *reinforcement learning* algorithms.

Video Content / Details of website for further learning (if any):
https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop
https://www.youtube.com/watch?v=ukzFI9rgwfU

Important Books/Journals for further learning including the page nos.:
Ethem Alpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 4-13

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

L - 3

LECTURE HANDOUTS

CSE

IV/VII - B

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning **Date of Lecture:**

Topic of Lecture: Supervised Learning: Learning a Class from Examples–Noise

Introduction: (Maximum 5 sentences) This is learning a class from its positive and negative examples. We generalize and discuss the case of multiple classes, then regression, where the outputs are continuous.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

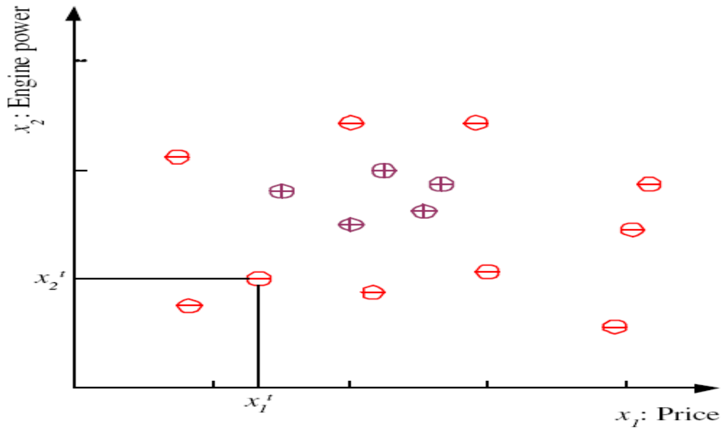
Concepts of Supervised Learning
Application of supervised learning

Detailed content of the Lecture:

*** Learning a Class from Examples**

- **Class C of a "family car"**
  - Prediction: Is car *x*a family car?
  - Knowledge extraction: What do people expect from a family car?
- **Output:**
  Positive (+) and negative (–) examples
- **Input representation:**
  x1: price, x2: engine power3

*** Training set X**



X ={***x**t*, *rt*}$^N_{t=1}$

*** Class C**

$(p1 \leq \text{price} \leq p2)$ AND $(e1 \leq \text{engine power} \leq e2)$

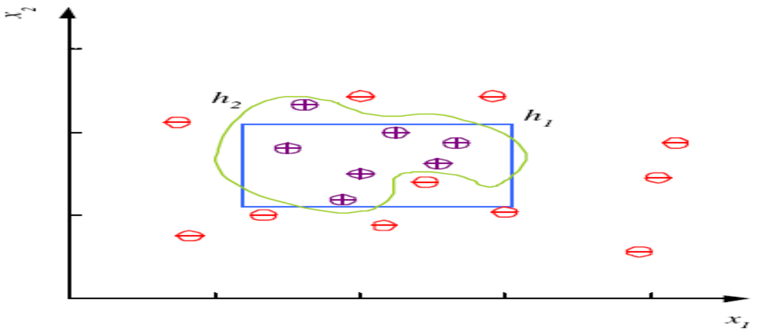
*** Hypothesis class H**

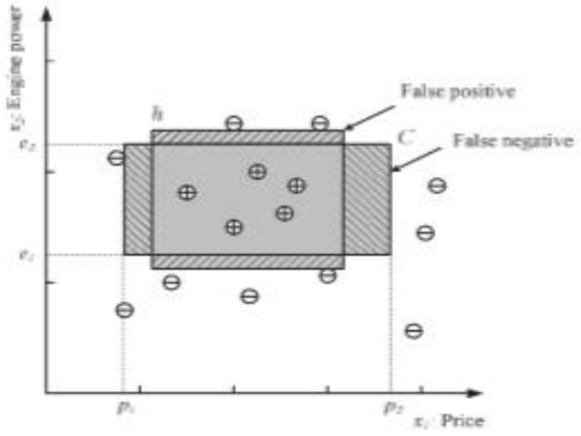
$$H(x)= \begin{cases} 1 \text{ if} h \text{ classifies } \boldsymbol{x} \text{ as a positive example} \\ 0 \text{ if} h \text{ classifies } \boldsymbol{x} \text{ as a negative example} \end{cases}$$

*** Noise and Model Complexity**

Use the simpler one because

- Simpler to use (lower computational complexity)
- Easier to train (lower space complexity)
- Easier to explain (more interpretable)
- Generalizes better (lower variance -Occam's razor)





If c is actual class and h is our induced hypothesis.The point where c is 1 but h is 0 is a false negative, and the point where c is0 and h is 1 is false positive.Other points-namely,true positives and true negatives – are correctly classified

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop

https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 22-32

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L - 4

LECTURE HANDOUTS

CSE

IV/VII - B

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning Date of Lecture:

Topic of Lecture: Learning Multiple Classes- Regression

Introduction: (Maximum 5 sentences) :

- In machine learning, multiclass or multinomial classification is the problem of classifying instances into one of three or more classes.
- Regression is used to predict the outcome of an event based on the relationship between variables obtained from the data-set. Linear regression is one type regression used in Machine Learning.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Learning a single class

Concepts of supervised learning

Detailed content of the Lecture:

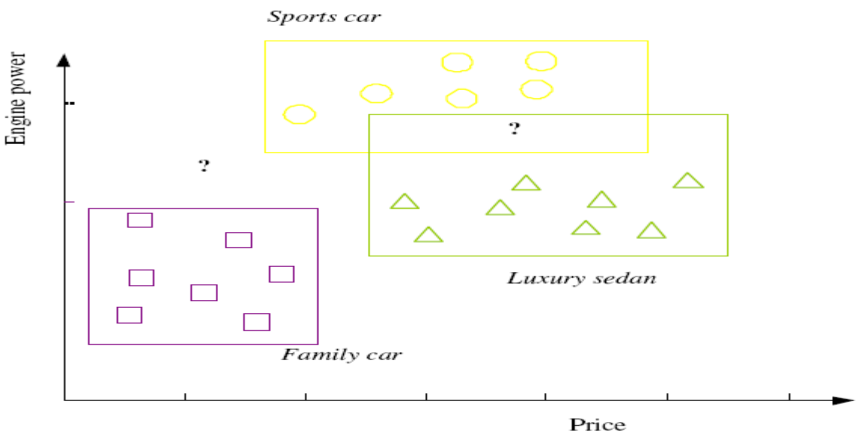
Multiple Classes, Ci i=1,...,K

- In our example of learning a family car, we have positive examples belonging to the class family car and the negative examples belonging to all other cars. This is a *two-class* problem.
- We have K classes denoted as C*i*, *i* = 1, . . . , *K*, and an input instance belongs to one and exactly one of them. The training set is now of the form

$$X = \{\boldsymbol{x}t , \boldsymbol{r}t\}^{N}_{t=1}$$

- where ***r*** has *K* dimensions and

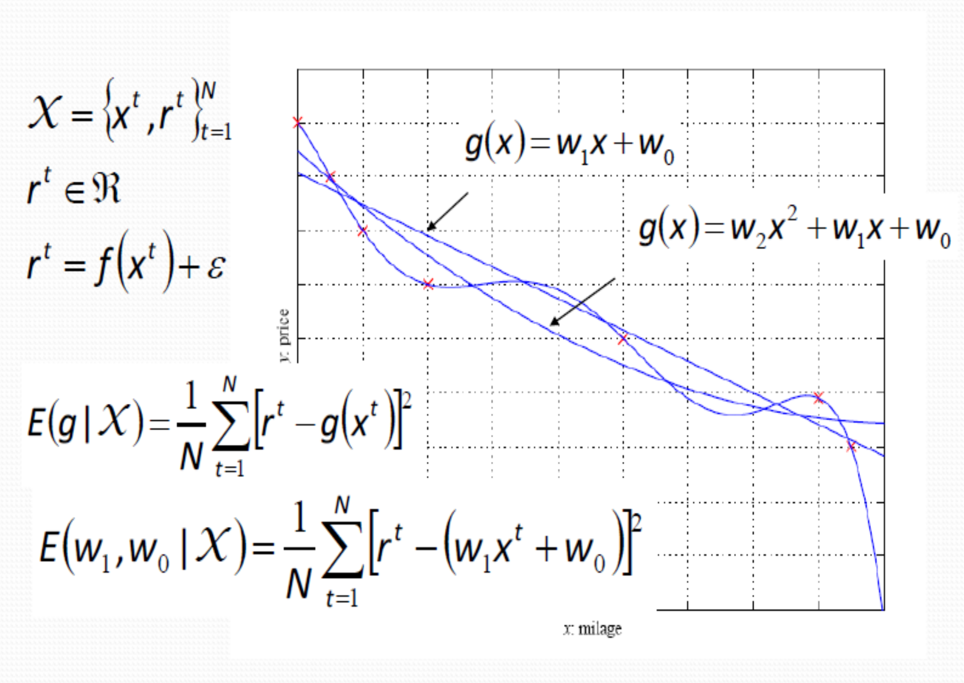
$$R^{t}_{i=}\begin{cases} 1 \text{ if } \boldsymbol{x}t \text{ €C}i \\ 0 \text{ if } \boldsymbol{x}t \text{ € C}j,\ j \neq i \end{cases}$$



- Train hypotheses *hi*(***x***), *i* =1,...,*K*:

$$h_i(\mathbf{x}^t) = \begin{cases} 1 \text{ if } \mathbf{x}^t \in C_i \\ 0 \text{ if } \mathbf{x}^t \in C_j, j \neq i \end{cases}$$

*** Regression**



$$\mathcal{X} = \{x^t, r^t\}_{t=1}^N$$

$$r^t \in \Re$$

$$r^t = f(x^t) + \varepsilon$$

$$E(g \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - g(x^t)\right]^2$$

$$E(w_1, w_0 \mid \mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\left[r^t - (w_1 x^t + w_0)\right]^2$$

Video Content / Details of website for further learning (if any):
https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop
https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 32-37

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

L - 5

LECTURE HANDOUTS

CSE **IV / VII-B**

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning Date of Lecture:

Topic of Lecture: Model Selection and Generalization

Introduction: (Maximum 5 sentences) : Model selection is the process of choosing between different machine learning approaches - e.g. SVM, logistic regression, etc - or choosing between different hyperparameters or sets of features for the same machine learning approach - e.g. deciding between the polynomial degrees/complexities for linear regression.
Generalization refers to your model's ability to adapt properly to new, previously unseen data, drawn from the same distribution as the one used to create the model.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Classification

Regression

Detailed content of the Lecture:

*** Model Selection & Generalization**

- ➢ Learning is an ill-posed problem; data is not sufficient to find a unique solution
- ➢ The need for inductive bias, assumptions about H
- ➢ Generalization: How well a model performs on new data
- ➢ Overfitting: Hmore complex than Cor *f*
- ➢ Underfitting: Hless complex than Cor *f*

*** Triple Trade-Off**

- ➢ There is a trade-off between three factors (Dietterich, 2003):

  1.Complexity of H, *c* (H),

  2.Training set size, *N*,

  3.Generalization error, E, on new data

- As N↑ ,E↓
- As c (H)↑ first E↓ and then ↑E

*** Cross-Validation**

- To estimate generalization error, we need data unseen during training. We split the data as
- Training set (50%)
- Validation set (25%)
- Test (publication) set (25%)
- Resampling when there is few data

*** Test Set**

- If we need to report the error to give an idea about the expected error of our best model, we should not use the vaslidation error.
- We have used the validation set to choose the best model, and it has eftest set fectively become a part of the training set.
- We need a third set, a *test set*, sometimes also called the *publication set*, containing examples not used in training or validation.
- An analogy from our lives is when we are taking a course:
    - the example problems that the instructor solves in class while teaching a subject form the training set;
    - exam questions are the validation set;
    - the problems we solve in our later, professional life are the

test set.

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop
https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 37- 41

Course Teacher

Verified by HOD

L - 6

LECTURE HANDOUTS

CSE

IV / VII-B

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning Date of Lecture:

Topic of Lecture: Bayesian Decision Theory: Classification-Examples

Introduction: (Maximum 5 sentences) : Bayes' Theorem is the fundamental result of probability theory – it puts the posterior probability P(H|D) of a hypothesis as a product of the probability of the data given the hypothesis(P(D|H)), multiplied by the probability of the hypothesis (P(H)), divided by the probability of seeing the data.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Concepts of Supervised Learning
Application of supervised learning
Probability and Inference

Detailed content of the Lecture:

.

*** Bayesian Decision theory**:

➢ Bayes' rule is used to calculate the probabilities of the classes. We generalize to discuss how we can make rational decisions among multiple actions to minimize expected risk. We also discuss learning association rules from data.

*** Two Roles for Bayesian Methods**

➢ **Provides practical learning algorithms**:

- Naive Bayes learning
- Bayesian belief network learning
- Combine prior knowledge (prior probabilities) with observed data
- Requires prior probabilities

- **Provides useful conceptual framework**
  - Provides gold standard" for evaluating other learning algorithm
  - Additional insight into Occam's razor

Bayes Theorem

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$$

- P (h) = prior probability of hypothesis h
- P (D) = prior probability of training data D
- P(h|D) = probability of h given D
- P(D|h) = probability of D given h

Choosing Hypotheses

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$$

Generally want the most probable hypothesis given the training data

Maximum a posteriori hypothesis h_{MAP} :

$$h_{MAP} = \underset{h2H}{\arg\max}\ P(D \mid h)P(h)$$

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop

https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 47-51

Course Teacher

Verified by HOD

L - 7

LECTURE HANDOUTS

CSE

IV / VII-B

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning **Date of Lecture:**

Topic of Lecture: Bayesian Decision Theory: Classification-Examples

Introduction: (Maximum 5 sentences) : In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Concepts of Supervised Learning

Bayes' rule

Detailed content of the Lecture:

*** Bayesian Decision theory**:

- ➢ Bayes' rule is used to calculate the probabilities of the classes. We generalize to discuss how we can make rational decisions among multiple actions to minimize expected risk. We also discuss learning association rules from data.

*** Classification Example:**

- ➢ We observe customer's yearly income and savings, which we represent by two random variables *X*1 and *X*2.
- ➢ It may again be claimed that if we had access to other pieces of knowledge such as the state of economy in full detail and full knowledge about the customer, his or her intention, moral codes, and so forth, whether someone is a low-risk or high-risk customer could have been deterministically calculated.
- ➢ But these are non observables and with what we can observe, the credibility of a customer is denoted by a Bernoulli random variable C conditioned on the observables ***X*** = *[X*1*,X*2*]T* where C = 1 indicates a high-risk customer and C = 0 indicates a low-risk customer.
- ➢ Thus if we know *P*(C|*X*1*,X*2*)*, when a new application arrives with *X*1 = *x*1 and *X*2 = *x*2, we can

$$\text{choose} \begin{cases} C = 1 \text{ if } P(C = 1 \mid x1, x2) > 0.5 \\ C = 0 \text{ otherwise} \end{cases}$$

or equivalently

$$\text{choose} \begin{cases} C = 1 \text{ if } P(C = 1 \mid x1, x2) > P(C = 0 \mid x1, x2) \\ C = 0 \text{ otherwise} \end{cases}$$

- The probability of error is 1 − max$(P(C = 1 | x1, x2), P(C = 0 | x1, x2))$.

*** Bayes rule:**

The problem Bayes' rule then is to be able to calculate $P(C | \boldsymbol{x})$. Using *Bayes' rule*, it can be written as

$$P(C|\mathbf{x}) = \frac{P(C)p(\mathbf{x}|C)}{P(x)}$$

*** Prior Probability:**

- $P(C = 1)$ is called the *prior probability* that C takes the value 1, which in our example corresponds to the probability that a customer is highrisk, regardless of the $\boldsymbol{x}$ value.
- It is called the prior probability because it is the knowledge we have as to the value of C *before* looking at the observables $\boldsymbol{x}$, satisfying

$$P(C = 0) + P(C =1) = 1$$

Class Likelihood:

- $P(\boldsymbol{x} | C)$ is called the *class likelihood* and is the conditional probability that an event belonging to C has the associated observation value $\boldsymbol{x}$.
- In our case, $p(x1, x2 | C = 1)$ is the probability that a high-risk customer has his or her $X1 = x1$ and $X2 = x2$. It is what the data tells us regarding the class.

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop

https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 47-51

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L - 8

LECTURE HANDOUTS

CSE

IV / VII-B

Course Name with Code : Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : I - Introduction and Supervised Learning **Date of Lecture:**

Topic of Lecture: Losses and Risks

Introduction: (Maximum 5 sentences) :

- A financial institution when making a decision for a loan applicant should take into account the potential gain and loss as well. An accepted low-risk applicant increases profit, while a rejected high-risk applicant decreases loss.
- The loss for a high-risk applicant erroneously accepted may be different from the potential gain for an erroneously rejected low-risk applicant.
- The situation is much more critical and far from symmetry in other domains like medical diagnosis or earthquake prediction.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Concepts of Supervised Learning

Application of supervised learning

Detailed content of the Lecture:

Losses and Risks

- It may be the case that decisions are not equally good or costly. A financial institution when making a decision for a loan applicant should take into account the potential gain and loss as well.
- An accepted low-risk applicant increases profit, while a rejected high-risk applicant decreases loss. The loss for a high-risk applicant erroneously accepted may be different from the potential gain for an erroneously rejected low-risk applicant.
- The situation is much more critical and far from symmetry in other domains like medical diagnosis or earthquake prediction.

Loss Function Expected Risk:

- Let us define action *ai* as the decision to assign the input to class C*i* loss function and *λik* as the *loss* incurred for taking action *ai* when the input actually belongs to Cexpected risk *k*. Then the *expected risk* for taking action *ai* is

$$R(ai\,|\,x) = \left\{ \begin{matrix} K & \lambda ikP(Ck\,|\,x) \\ \Sigma & \end{matrix} \right. \qquad \lambda_{ik} = \begin{cases} 0 \text{ if } i = k \\ 1 \text{ if } i \neq k \end{cases}$$

$k=1$

and we choose the action with minimum risk:

choose ai if

$$R(ai\,|\,\mathbf{x}) = \min_{k} R(ak\,|\,\mathbf{x})$$

Reject:

- In such a case, we define reject an additional action of *reject* or *doubt*, $aK+1$, with ai, $i = 1, \ldots, K$, being the usual actions of deciding on classes Ci, $i = 1, \ldots, K$ (Duda, Hart, and Stork 2001).
- A possible loss function is

$$\lambda ik = \begin{cases} 0 \text{ if} i = k \\ \lambda \text{ if } i = K + 1 \\ 1 \text{ otherwise} \end{cases}$$

Where $0 < \lambda < 1$ is the loss incurred for choosing the $(K + 1)$st action of reject. Then the risk of reject is

$$R(aK+1\,|\,\mathbf{x}) = \sum_{k=1}^{K} \lambda P(Ck\,|\,\mathbf{x}) = \lambda$$

- The optimal decision rule is to

choose Ci if

$$R(\alpha i\,|\,\mathbf{x}) < R(\alpha k\,|\,\mathbf{x}) \text{ for all } k \neq i \text{ and}$$
$$R(\alpha i\,|\,\mathbf{x}) < R(\alpha K+1\,|\,\mathbf{x})$$

reject if

$$R(\alpha K+1\,|\,\mathbf{x}) < R(\alpha i\,|\,\mathbf{x}),\ i = 1, \ldots, K$$

Given the loss function of equation this simplifies to

choose Ci if

$$P(Ci\,|\,\mathbf{x}) > P(Ck\,|\,\mathbf{x}) \text{ for all } k \neq i \text{ and}$$
$$P(Ci\,|\,\mathbf{x}) > 1 - \lambda$$

reject otherwise .

This whole approach is meaningful if $0 < \lambda < 1$:

If $\lambda = 0$, we always reject; a reject is as good as a correct classification.

If $\lambda \geq 1$, we never reject; a reject is as costly as, or costlier than, an error.

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop

https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 51-53

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

L - 9

LECTURE HANDOUTS

CSE **IV / VII-B**

Course Name with Code : Machine Learning Techniques -16CSE14
Course Teacher : Dr.N.Naveenkumar
Unit : I - Introduction and Supervised Learning **Date of Lecture:**

Topic of Lecture: Discriminant Functions & Association rules

Introduction: (Maximum 5 sentences) :

- A function of several variates used to assign items into one of two or more groups.
- The function for a particular set of items is obtained from measurements of the variates of items which belong to a known group. It is called Discriminant Function.
- Association rule mining finds interesting associations and relationships among large sets of data items.
- This rule shows how frequently a item set occurs in a transaction. A typical example is Market Based Analysis.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Concepts of Supervised Learning

Application of supervised learning

Detailed content of the Lecture:

Classification can also be seen as implementing a set of discriminant functions , gi(x),i =1,...,K, such that we

choose Ci if gi(x)=max k gk(x)

We can represent the Bayes' classifier in this way by setting

gi(x)=−R(αi | x)

and the maximum discriminant function corresponds to minimum conditional risk. When we use the 0/1 loss function, we have

gi(x)= P(Ci | x)

or ignoring the common normalizing term, p(x), we can write

gi(x)= p(x | Ci)P(Ci)

This divides the feature space into K R1,...,RK, wheredecision regions Ri ={ x | gi(x) = maxk gk(x)}. The regions are separated by decision boundaries, surfaces in feature space where ties occur among the largest discriminant functions (see figure).

When there are two classes, we can define a single discriminant

g(x)=g1(x)−g2(x) and we

$$\text{choose} = \begin{cases} \text{C1 if } g(x)>0 \\ \text{C2 otherwise} \end{cases}$$

An example is a two-class learning problem where the positive examples can be taken as C1 and the negative examples as C2. WhenK = 2, the classification system is a dichotomizer and for K ≥ 3, it is a poly-dichotomizer.

- An association rule is an implication of the form X → Y where X is the antecedent and Y is the consequent of the rule. One example of association rules is in basket analysis where we want to find the dependency between two items X and Y. The typical application is in retail where

X and Y are items sold, In learning association rules, there are three measures that are frequently calculated

Support of the association rule X →Y:

$$\text{Support}(X,Y)\equiv P(X,Y)=\frac{\#\{\text{customers who bought X and Y}\}}{\#\{\text{customers}\}}$$

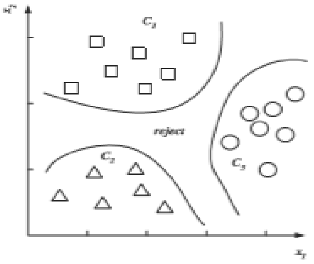


Figure 3.1 Example of decision regions and decision boundaries.

Confidence of the association rule X →Y:

$$\text{Confidence}(X \rightarrow Y)\equiv P(Y\,|\,X) = \frac{P(X,Y)}{P(X)}$$

$$= \frac{\#\{\text{customers who bought X and Y}\}}{\#\{\text{customers who bought X}\}}$$

Lift, also known asinterest of the association rule X →Y}

$$\text{Lift}(X \rightarrow Y)= \frac{P(X,Y)}{P(X)P(Y)} = \frac{P(Y\,|\,X)}{P(Y)}$$

Apriori algorithm:

1.To find frequent itemsets quickly (without complete enumeration of all subsets of items), the Apriori algorithm uses the fact that for{X,Y,Z} to be frequent (have enough support), all its subsets {X,Y}, {X,Z}, and{Y,Z} should be frequent as well—adding another item can never increase support.

* That is, we only need to check for three-item sets all of whose two-item subsets are frequent; or, in other words, if a twoitem set is known not to be frequent, all its supersets can be pruned and need not be checked.

* We start by finding the frequent one-item sets and at each step, inductively, from frequent k-item sets, we generate candidate k+1-item sets and then do a pass over the data to check if they have enough support.

* The Apriori algorithm stores the frequent itemsets in a hash table for easy access.

Note that the number of candidate itemsets will decrease very rapidly as k increases. If the largest itemset has n items, we need a total of n+1 passes over the data.

2. Once we find the frequent k-item sets, we need to convert them to rules by splitting the k items into two as antecedent and consequent.

* Just like we do for generating the itemsets, we start by putting a single consequent and k−1 items in the antecedent.

* Then, for all possible single consequents, we check if the rule has enough confidence and remove it if it does not.

* Here, as in itemset generation, we use the fact that to be able to have rules with two items in the consequent with enough confidence, each of the two rules with single consequent by itself should have enough confidence; that is, we go from one consequent rules to two consequent rules and need not check for all possible two-term consequents

Video Content / Details of website for further learning (if any):

https://lecturenotes.in/notes/24274-note-for-machine-learning-ml-by-new-swaroop

https://www.youtube.com/watch?v=WpxK__SK2a0

Important Books/Journals for further learning including the page nos.:

Ethem Alpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

DESIGNING YOUR FUTURE
Estd. 2000

LECTURE HANDOUTS

L - 1

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **II** **Date of Lecture:**

Topic of Lecture: Parametric Classification–Regression

Introduction: (Maximum 5 sentences):A learning model that summarizes data with a set of parameters of fixed size (independent of the number of training examples) is called a parametric model.

The algorithms involve two steps:

- Select a form for the function.
- Learn the coefficients for the function from the training data.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Machine Learning Techniques(Classification, Regression)

Implementation Of Algorithms

Detailed content of the Lecture:

- Combining the prior and what the data tells us using Bayes' rule, we the posterior probability of the concept, P(C|x),
- posterior = prior × likelihood /evidence.
- posterior probability of class Ci as

$$\mathrm{P(Ci|x)} = \mathrm{p(x|Ci)P(Ci)/p(x)} = \mathrm{p(x|Ci)P(Ci)}/\sum_{k=1}^{k} p\,\mathrm{(x|Ck)P(Ck)}$$

- Mean Variance

$$m_i = \frac{\sum_t x^t r_i^t}{\sum_t r_i^t} \quad s_i^2 = \frac{\sum_t (x^t - m_i)^2 r_i^t}{\sum_t r_i^t}$$

- This is the likelihood-based approach to classification where we use data to estimate the densities separately, calculate posterior densities using Bayes' rule, and then get the discriminant.

$$x = \frac{m_1 + m_2}{2}$$

- In regression, we would like to write the numeric output, called the dependent variable, as a function of the input, called the independent variable.

$r = f(x) + \epsilon$ f (x) is the unknown function.

- In linear regression, we have a linear model

$$\sum_t r^t = N w_0 + w_1 \sum_t x^t$$

$$\sum_t r^t x^t = w_0 \sum_t x_t + w_1 \sum_t (x^t)^2$$

which can be written in vector-matrix form as $\mathbf{A}\boldsymbol{w} = \boldsymbol{y}$ where

$$\mathbf{A} = \begin{bmatrix} N & \sum_t x^t \\ \sum_t x^t & \sum_t (x^t)^2 \end{bmatrix}, \boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} \sum_t r^t \\ \sum_t r^t x^t \end{bmatrix}$$

In the general case of polynomial regression, the model is a polynomial regression in x of order k

g(xt |wk,...,w2, w1, w0) = wk(xt)k +···+ w2(xt)2 + w1xt + w0.

.

- vector matrix form Aw = y

$$\mathbf{A} = \begin{bmatrix} N & \sum_t x^t & \sum_t (x^t)^2 & \cdots & \sum_t (x^t)^k \\ \sum_t x^t & \sum_t (x^t)^2 & \sum_t (x^t)^3 & \cdots & \sum_t (x^t)^{k+1} \\ \vdots & & & & \\ \sum_t (x^t)^k & \sum_t (x^t)^{k+1} & \sum_t (x^t)^{k+2} & \cdots & \sum_t (x^t)^{2k} \end{bmatrix}$$

$$\boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_k \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} \sum_t r^t \\ \sum_t r^t x^t \\ \sum_t r^t (x^t)^2 \\ \vdots \\ \sum_t r^t (x^t)^k \end{bmatrix}$$

Benefits of Parametric Machine Learning Algorithms:

- **Simpler**: These methods are easier to understand and interpret results.
- **Speed**: Parametric models are very fast to learn from data.

Video Content / Details of website for further learning (if any):

https://machinelearningmastery.com/parametric-and-nonparametric-machine-learning-algorithms/

https://youtu.be/NEcCfM2a3n0.

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 69-75

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 2

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**
Unit : **IIDate of Lecture:**

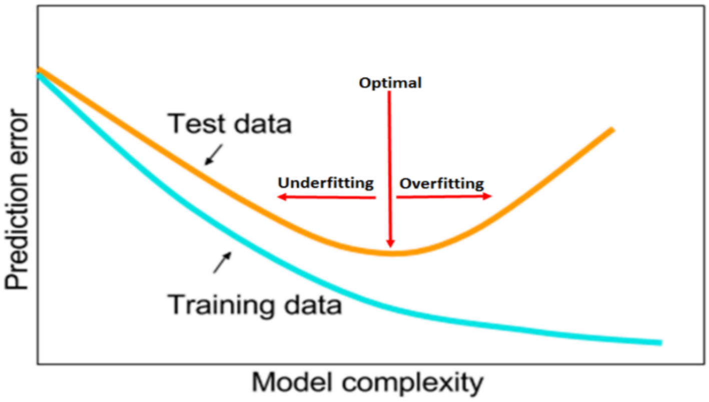
Topic of Lecture: Tuning Model Complexity–Model Selection Procedures

Introduction: (Maximum 5 sentences): **Tuning** is the process of maximizing a **model's** performance without overfitting or creating too high of a variance.**Model selection** is the **process of selecting** one final **machine learning model** from among a collection of candidate **machine learning models** for a training dataset

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Parametric Classification–Regression
What is model .

Detailed content of the Lecture:

- Hyperparameters differ from other **model** parameters in that they are not learned by the **model** automatically through training **methods**.
- In **machine learning**, **model complexity** often refers to the number of features or terms included in a given predictive **model**, as well as whether the chosen **model** is linear, nonlinear, and so on. It can also refer to the algorithmic **learning complexity** or computational **complexity**.
- When you **increase complexity** of **your model**, it is more likely to overfit, meaning it will adapt to training data very well, but will not figure out general relationships in **the** data. In such case, performance on **a** test set is going to be poor. ... This leads to poor test set performance.



- **Model selection** is the **process of** choosing between different **machine learning** approaches - e.g. SVM, logistic regression, etc - or choosing between different hyperparameters or sets of features for the same **machine learning** approach - e.g. deciding between the polynomial degrees/complexities for linear regression

- The choice of the actual machine learning algorithm (e.g. SVM or logistic regression) is less important than you'd think - there may be a "best" algorithm for a particular problem, but often its performance is not much better than other well-performing approaches for that problem.

There may be certain qualities you look for in an model:

- Interpretable - can we see or understand why the model is making the decisions it makes?
- Simple - easy to explain and understand
- Accurate
- Fast (to train and test)
- Scalable (it can be applied to a large dataset)

Though there are generally trade-offs amongst these qualities.

There are a number of procedures we can use to fine-tune model complexity.

- **cross-validation**

  In practice, the method we use to find the optimal complexity is cross- validation. We cannot calculate bias and variance for a model, but we can calculate the total error.

- **Regularization**

  Another approach that is used frequently is regularization,
  In this approach, we write an augmented error function
  E' =error on data + λmodel complexity.

Video Content / Details of website for further learning (if any):
https://youtu.be/VZuKBKd4ck4

https://www.innoarchitech.com/blog/machine-learning-an-in-depth-non-technical-guide-part-3

Important Books/Journals for further learning including the page nos.:
EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 76-84

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

DESIGNING YOUR FUTURE
Estd. 2000

IQAC

LECTURE HANDOUTS

L - 3

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : IIDate of Lecture:

Topic of Lecture: Multivariate Methods: Data–Parameter Estimation

Introduction: (Maximum 5 sentences): **Multivariate data** is the **data** in which **analysis** are based on more than two variables per observation. Usually **multivariate data** is used for explanatory purposes. In many applications, several measurements are made on each individual or event generating an observation vector. The sample may be viewed as a data matrix.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Machine algorithms
Analysis of problem.

Detailed content of the Lecture:

- **Univariate data** is used for the simplest form of analysis. It is the type of data in which analysis are made only based on one variable.

- **Bivariate data** is used for little complex analysis than as compared with univariate data. Bivariate data is the data in which analysis are based on two variables per observation simultaneously.

Multivariate data viewed as a data matrix.

$$X= \begin{bmatrix} X_1^1 & X_2^1 & . & . & . & X_d^1 \\ X_1^2 & X_2^2 & . & . & . & . \\ . & . & & & & . \\ . & . & & & & \\ X_1^N & . & . & . & . & X_d^N \end{bmatrix}$$

.

- d variables denoting the result of measurements made on an individual or event. N rows correspond to independent and feature attribute identically distributed observations

mean vector

The mean vector μ is defined such that each of its elements is the mean of one column of X:

$$E[\mathbf{x}] = \boldsymbol{\mu} = [\mu_1, \ldots, \mu_d]^T$$

The variance of X_i is denoted as σ_2^i, and the covariance of two variables X_i and $X_j$is defined as

$$\sigma_{ij} \equiv \mathrm{Cov}(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)] = E[X_i X_j] - \mu_i \mu_j$$

covariance matrix

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & & & \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix}$$

In vector-matrix notation

$$\Sigma \equiv \mathrm{Cov}(\mathbf{X}) = E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T] = E[\mathbf{X}\mathbf{X}^T] - \boldsymbol{\mu}\boldsymbol{\mu}^T$$

* correlation

The correlation between variables X_i and X_j is a statistic normalized between −1 and+1, defined as

$$\mathrm{Corr}(X_i, X_j) \equiv \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$$

* sample mean

$$\mathbf{m} = \frac{\sum_{t=1}^{N} \mathbf{x}^t}{N} \text{ with } m_i = \frac{\sum_{t=1}^{N} x_i^t}{N}, i = 1, \ldots, d$$

* sample covariance

$$s_i^2 = \frac{\sum_{t=1}^{N} (x_i^t - m_i)^2}{N}$$

$$s_{ij} = \frac{\sum_{t=1}^{N} (x_i^t - m_i)(x_j^t - m_j)}{N}$$

* sample correlation

$$r_{ij} = \frac{s_{ij}}{s_i s_j}$$

Video Content / Details of website for further learning (if any):

https://youtu.be/IkvwXPEBlNo

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5742591/

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 87-89

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 4

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : IIDate of Lecture:

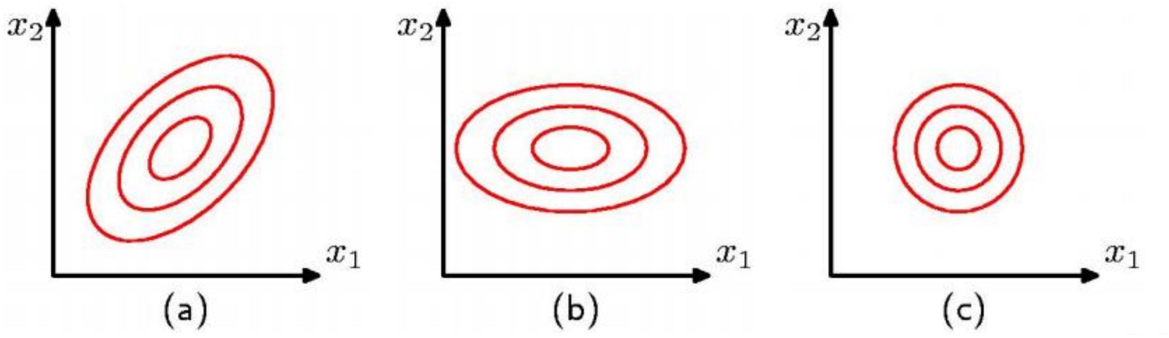
Topic of Lecture: Estimation of Missing Values–Multivariate Normal Distribution
Introduction: (Maximum 5 sentences): Frequently, values of certain variables may be missing in observations. The best strategy is to discard those observations all together, but generally we do not have large enough samples to be able to afford this and we do not want to lose data as the non-missing entries do contain information. We try to fill in the missing entries by estimating them. This is called **imputation.**
Prerequisite knowledge for Complete understanding and learning of Topic: **(Max. Four important topics)** Parametric Classification–Regression What is model .

Detailed content of the Lecture:

- **mean imputation**, for a numeric variable, we substitute the mean (average) of the available data for that variable in the sample.
- In **imputation by regression**, we try to predict the value of a missing variable from other variables whose values are known for that case.

Multivariate Normal Distribution

- A **multivariate normal distribution** is a vector in multiple normally **distributed** variables, such that any linear combination of the variables is also normally **distributed**.
- The **multivariate normal distribution** is a multidimensional generalization of the one-dimensional **normal distribution** . It represents the **distribution** of a **multivariate** random variable that is made up of multiple random variables that can be correlated with each other.



$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\boldsymbol{\Sigma}|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Video Content / Details of website for further learning (if any):

https://youtu.be/fm5TVg0wTB0

https://brilliant.org/wiki/multivariate-normal-distribution/

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 89-93.

Course Teacher

Verified by HOD

Estd. 2000

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

LECTURE HANDOUTS

L - 5

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14
Course Teacher :Dr.N.Naveenkumar
Unit : II
Date of Lecture:

Topic of Lecture: Multivariate Classification and Regression

Introduction: (Maximum 5 sentences): A class or cluster is a grouping of points in this multidimensional attribute space. Two locations belong to the same class or cluster if their attributes (vector of band values) are similar. A multiband raster and individual single band rasters can be used as the input into a **multivariate** statistical analysis.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Parametric Classification–Regression - Multi variate method.

When x ∈d, if the class-conditional densities, p(x|Ci), are taken as normal density, Nd(μi,Σi), we have

$$p(\boldsymbol{x}|C_i) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_i|^{1/2}}\exp\left[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right]$$

we define the discriminant function as

$$g_i(\boldsymbol{x}) = \log p(\boldsymbol{x}|C_i) + \log P(C_i)$$

and assuming $p(\boldsymbol{x}|C_i) \sim \mathcal{N}_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, we have

$$g_i(\boldsymbol{x}) = -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\boldsymbol{\Sigma}_i| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) + \log P(C_i)$$

Given a training sample for K ≥ 2 classes, X={ x^t,r^t}, where r^t_i= 1 if x^t∈C_i and 0 otherwise, estimates for the means and covariances are found using maximum likelihood separately for each class:

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}$$

$$\boldsymbol{m}_i = \frac{\sum_t r_i^t \boldsymbol{x}^t}{\sum_t r_i^t}$$

$$\mathbf{S}_i = \frac{\sum_t r_i^t(\boldsymbol{x}^t-\boldsymbol{m}_i)(\boldsymbol{x}^t-\boldsymbol{m}_i)^T}{\sum_t r_i^t}$$

These are then plugged into the discriminant function to get the estimates for the discriminants. Ignoring the first constant term, we have

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}(\boldsymbol{x}-\boldsymbol{m}_i)^T\mathbf{S}_i^{-1}(\boldsymbol{x}-\boldsymbol{m}_i) + \log\hat{P}(C_i)$$

Expanding this, we get

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}\left(\boldsymbol{x}^T\mathbf{S}_i^{-1}\boldsymbol{x} - 2\boldsymbol{x}^T\mathbf{S}_i^{-1}\boldsymbol{m}_i + \boldsymbol{m}_i^T\mathbf{S}_i^{-1}\boldsymbol{m}_i\right) + \log\hat{P}(C_i)$$

which defines a quadratic discriminant (see figure 5.3) that can also be written as

$$g_i(\boldsymbol{x}) = \boldsymbol{x}^T\mathbf{W}_i\boldsymbol{x} + \boldsymbol{w}_i^T\boldsymbol{x} + w_{i0}$$

Where

$$\begin{aligned} \mathbf{W}_i &= -\frac{1}{2}\mathbf{S}_i^{-1} \\ \boldsymbol{w}_i &= \mathbf{S}_i^{-1}\boldsymbol{m}_i \\ w_{i0} &= -\frac{1}{2}\boldsymbol{m}_i^T\mathbf{S}_i^{-1}\boldsymbol{m}_i - \frac{1}{2}\log|\mathbf{S}_i| + \log\hat{P}(C_i) \end{aligned}$$

Another possibility is to pool the data and estimate a common covariance matrix for all classes:

$$\mathbf{S} = \sum_i \hat{P}(C_i)\mathbf{S}_i$$

In this case of equal covariance matrices, reduces to

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m}_i)^T\mathbf{S}^{-1}(\boldsymbol{x}-\boldsymbol{m}_i) + \log\hat{P}(C_i)$$

linear discriminant

the quadratic term $x^T S^{-1} x$ cancels since it is common in all discriminants, and the decision boundaries are linear, leading to a linear discriminant that can linear discriminant be written as

$$g_i(\boldsymbol{x}) = \boldsymbol{w}_i^T\boldsymbol{x} + w_{i0}$$

Where

where

$$\begin{aligned} \boldsymbol{w}_i &= \mathbf{S}^{-1}\boldsymbol{m}_i \\ w_{i0} &= -\frac{1}{2}\boldsymbol{m}_i^T\mathbf{S}^{-1}\boldsymbol{m}_i + \log\hat{P}(C_i) \end{aligned}$$

Naive Bayes' classifier

the naive Bayes' classifier where p(xj|Ci) are univariate Gaussian. S and its inverse are diagonal, and we get

$$g_i(\boldsymbol{x}) = -\frac{1}{2}\sum_{j=1}^{d}\left(\frac{x_j^t - m_{ij}}{s_j}\right)^2 + \log\hat{P}(C_i)$$

Euclidean distance

Simplifying even further, if we assume all variances to be equal, the Mahalanobis distance reduces to Euclidean distance

$$g_i(\boldsymbol{x}) = -\frac{\|\boldsymbol{x}-\boldsymbol{m}_i\|^2}{2s^2} + \log\hat{P}(C_i) = -\frac{1}{2s^2}\sum_{j=1}^{d}(x_j^t - m_{ij})^2 + \log\hat{P}(C_i)$$

All classes have equal, diagonal covariance matrices, but variances are not equal.

Multivariate Regression

The goal in any data analysis is to extract from raw information the accurate estimation.

... **Multivariate Regression** is one of the simplest **Machine Learning** Algorithm. It comes under the class of Supervised **Learning** Algorithms i.e, when we are provided with training dataset.

In multivariate linear regression, the numeric output r is assumed to be written as a linear function, that is, a weighted sum, of several input variables, $x_1,\ldots,x_d$, and noise. Actually in statistical literature, this is called multiple regression; statisticians use the term multivariate when there are multiple outputs. The multivariate linear model is

$$r^t = g(x^t|w_0,w_1,\ldots,w_d) + \epsilon = w_0 + w_1x_1^t + w_2x_2^t + \cdots + w_dx_d^t + \epsilon$$

As in the univariate case, we assume to be normal with mean 0 and constant variance, and maximizing the likelihood is equivalent to minimizing the sum of squared errors:

$$E(w_0,w_1,\ldots,w_d|X) = \frac{1}{2}\sum_t (r^t - w_0 - w_1x_1^t - w_2x_2^t - \cdots - w_dx_d^t)^2$$

Let us define the following vectors and matrix:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_d^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & & & & \\ 1 & x_1^N & x_2^N & \cdots & x_d^N \end{bmatrix}, \boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d, \end{bmatrix}, \boldsymbol{r} = \begin{bmatrix} r^1 \\ r^2 \\ \vdots \\ r^N \end{bmatrix}$$

Then the normal equations can be written as

$$\mathbf{X}^T\mathbf{X}\boldsymbol{w} = \mathbf{X}^T\boldsymbol{r}$$

and we can solve for the parameters as

$$\boldsymbol{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{r}$$

Video Content / Details of website for further learning (if any):

1.https://youtu.be/6za9_mh3uTE 2. https://www.geeksforgeeks.org/multivariate-regression/

Important Books/Journals for further learning including the page nos.: .

EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 94-105

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 6

CSE

IV/VII - B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: II** **Date of Lecture:**

Topic of Lecture: Semi parametric method: Clustering k–Means Clustering

Introduction: (Maximum 5 sentences): A **semiparametric** model is a regression model with both a finite- and an infinite-dimensional component. ... Infinite dimensional spaces are spaces that have an infinite, and possibly ill-**defined**, number of dimensions and possibilities. They aren't spanned by any finite list of vectors.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Parametric Classification–Regression
Clustering

- **Clustering** is one of the most common exploratory data analysis technique used to get an intuition about the structure of the data.
- It can be defined as the task of identifying subgroups in the data such that data points in the same subgroup (cluster) are very similar while data points in different clusters are very different.
- Clustering is used in market segmentation; where we try to fined customers that are similar to each other whether in terms of behaviors or attributes, image segmentation/compression; where we try to group similar regions together, document clustering based on topics, etc.

K-means Algorithm

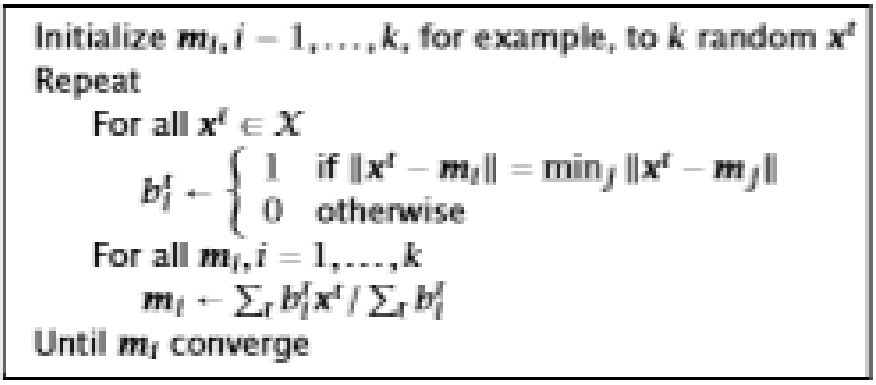
- K-means clustering is a very popular unsupervised learning algorithm.
- **K-means** algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to **only one group**.
- K-Means Clustering is an algorithm that, given a dataset, will identify which data points belong

to each one of the *k* clusters. It takes your data and learns how it can be grouped.

- Through a series of iterations, the algorithm creates groups of data points — referred to as clusters — that have similar variance and that minimize a specific cost function: the within-cluster sum of squares.

$$arg_C min \sum_{i=1}^{k} \sum_{x \in C_i} \| x - \mu_i \|^2$$

K-MEANS CLUSTERING ALGORITHM



```
Initialize $m_i, i = 1,\dots,k$, for example, to $k$ random $x^t$
Repeat
    For all $x^t \in \mathcal{X}$
        $b_i^t \leftarrow \begin{cases} 1 & \text{if } \|x^t - m_i\| = \min_j \|x^t - m_j\| \\ 0 & \text{otherwise} \end{cases}$
    For all $m_i, i = 1,\dots,k$
        $m_i \leftarrow \sum_t b_i^t x^t / \sum_t b_i^t$
Until $m_i$ converge
```

Applications

k-means algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc. The goal usually when we undergo a cluster analysis is either:

- Get a meaningful intuition of the structure of the data we're dealing with.
- Cluster-then-predict where different models will be built for different subgroups if we believe there is a wide variation in the behaviors of different subgroups. An example of that is clustering patients into different subgroups and build a model for each subgroup to predict the probability of the risk of having heart attack.

Video Content / Details of website for further learning (if any):
https://youtu.be/2_TQPwUkcas

https://www.statisticshowto.datasciencecentral.com/semiparametric/

Important Books/Journals for further learning including the page nos.: .

EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 143-149

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

LECTURE HANDOUTS

L - 7

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: II** **Date of Lecture:**

Topic of Lecture: Hierarchical Clustering

Introduction: (Maximum 5 sentences): In data mining and statistics, **hierarchical clustering** (also called **hierarchical cluster analysis** or **HCA**) is a method of cluster analysis which seeks to build a hierarchy of clusters.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Parametric Classification–Regression
Clustering.

Hierarchical clustering generally fall into two types:

- **Agglomerative**: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
- **Divisive**: This is a "top-down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

➢ In general, the merges and splits are determined in a greedy manner. The results of hierarchical clustering are usually presented in a dendrogram.

At each iteration of an agglomerative algorithm, we choose the two closest groups to merge.

single-link clustering

In single-link clustering, this distance is defined as the smallest distance between all possible pair of elements of the two groups:

$$d(\mathcal{G}_i,\mathcal{G}_j) = \min_{\boldsymbol{x}^r\in\mathcal{G}_i,\boldsymbol{x}^s\in\mathcal{G}_j} d(\boldsymbol{x}^r,\boldsymbol{x}^s)$$

complete-link clustering

In complete-link clustering, the distance between two groups is taken as the largest distance between all possible pairs:

$$d(\mathcal{G}_i,\mathcal{G}_j) = \max_{\boldsymbol{x}^r\in\mathcal{G}_i,\boldsymbol{x}^s\in\mathcal{G}_j} d(\boldsymbol{x}^r,\boldsymbol{x}^s)$$

Video Content / Details of website for further learning (if any):
https://youtu.be/7enWesSofhg

https://www.displayr.com/what-is-hierarchical-clustering/

Important Books/Journals for further learning including the page nos.: .

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 157-158

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 8

CSE

IV/VII- B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: II** **Date of Lecture:**

Topic of Lecture: Hierarchical Clustering-example problem

Introduction: (Maximum 5 sentences): In data mining and statistics, **hierarchical clustering** (also called **hierarchical cluster analysis** or **HCA**) is a method of cluster analysis which seeks to build a hierarchy of clusters.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Parametric Classification–Regression

Hierarchical Clustering Concept

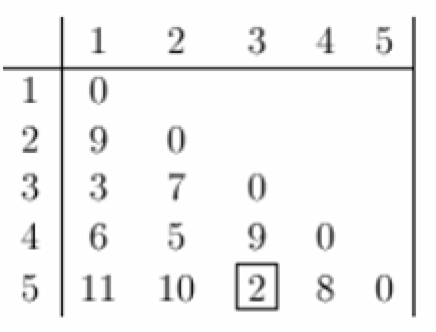
Hierarchical clustering generally fall into two types:

- **Agglomerative**: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

Agglomerative Hierarchical Clustering

Example of Complete Linkage Clustering

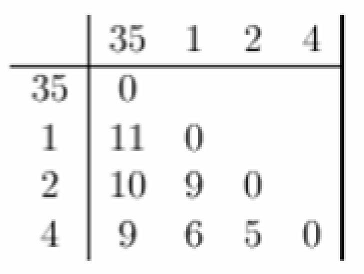
- Clustering starts by computing a distance between every pair of units that you want to cluster. A distance matrix will be symmetric (because the distance between x and y is the same as the distance between y and x) and will have zeroes on the diagonal (because every item is distance zero from itself). The table below is an example of a distance matrix.



	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	$\boxed{2}$	8	0

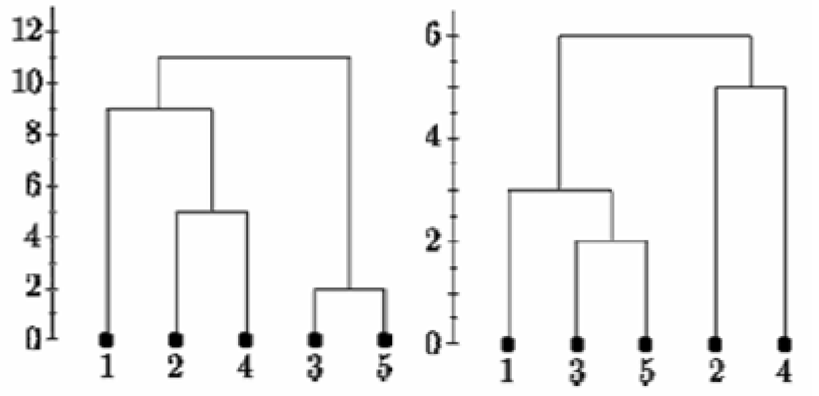
The smallest distance is between three and five and they get linked up or merged first into a the cluster '35'.

- To obtain the new distance matrix, we need to remove the 3 and 5 entries, and replace it by an entry "35" . Since we are using complete linkage clustering, the distance between "35" and every other item is the maximum of the distance between this item and 3 and this item and 5. For example, d(1,3)= 3 and d(1,5)=11. So, D(1,"35")=11. This gives us the new distance matrix. The items with the smallest distance get clustered next. This will be 2 and 4.



	35	1	2	4
35	0			
1	11	0		
2	10	9	0	
4	9	6	5	0

- Continuing in this way, after 6 steps, everything is clustered. This is summarized below. On this plot, the y-axis shows the distance between the objects at the time they were clustered. This is called the cluster height. Different visualizations use different measures of cluster height.



Complete linkage **Single linkage**

Video Content / Details of website for further learning (if any):
https://youtu.be/2MJQyh__kpc

https://people.revoledu.com/kardi/tutorial/Clustering/Numerical%20Example.html

Important Books/Journals for further learning including the page nos.: .

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no : 149-150

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 9

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : II

Date of Lecture:

Topic of Lecture:K–Means Clustering- example problem

Introduction: (Maximum 5 sentences): **Clustering** is one of the most common exploratory data analysis technique used to get an intuition about the structure of the data.It can be defined as the task of identifying subgroups in the data such that data points in the same subgroup (cluster) are very similar while data points in different clusters are very different.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Parametric Classification–Regression
K-Means Clustering Concepts

K Means Clustering

- K-means clustering is a very popular unsupervised learning algorithm.
- **K-means** algorithm is an iterative algorithm that tries to partition the dataset into *K*pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to **only one group**.

K Means Clustering Algorithm:

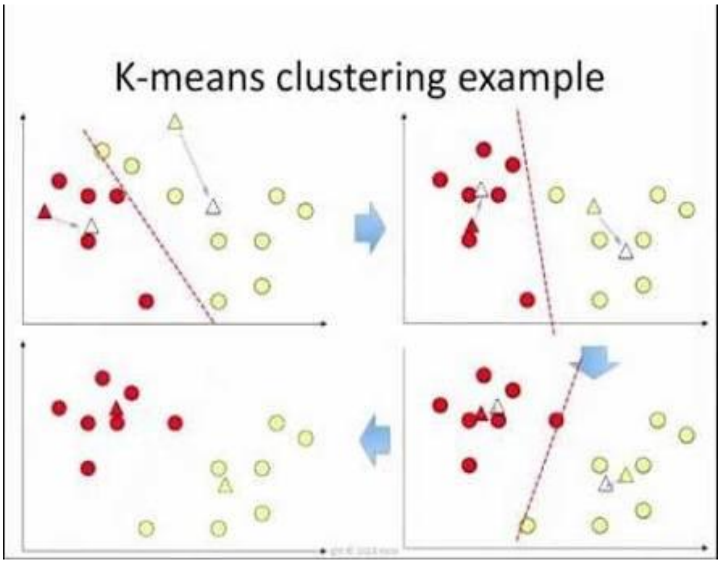
- **Step 1:** Visualize n data points and decide the number of clusters (k). Choose k random points on the graph as the centroids of each cluster. For this example, we would like to divide the data into 4 clusters, so we pick 4 random centroids.
- **Step 2:** Calculate the Euclidean distance between each data point and chosen clusters' centroids. A point is considered to be in a particular cluster if it is closer to that cluster's
- **Step 3:** After assigning all observations to the clusters, calculate the clustering score, by summing up all the Euclidean distances between each data point and the corresponding centroid.

$$\text{Total distances} = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

Where:
k: the number of clusters
n: the number of points belonging to cluster j
cj: the centroid of cluster j

- **Step 4:** Define the new centroid of each cluster by calculating the mean of all points assigned to that cluster. Here's the formula (n is the number of points assigned to that cluster)

- **Step 5:** Repeat from step 2 until the positions of the centroids no longer move and the he assignments stay the same.



Video Content / Details of website for further learning (if any):
https://youtu.be/YWgcKSa_2ag

https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a

Important Books/Journals for further learning including the page nos.: .

EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no : 158-159

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L -1

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : III

Date of Lecture:

Topic of Lecture: Introduction-Neural Network representation

Introduction: (Maximum 5 sentences): An **artificial neural network** (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Artificial neural network

Neural network

Artificial neural network (ANN):

- ANNs are considered nonlinear statistical data modeling tools where the complex relationships between inputs and outputs are modeled or patterns are found.ANN is also known as a neural network.

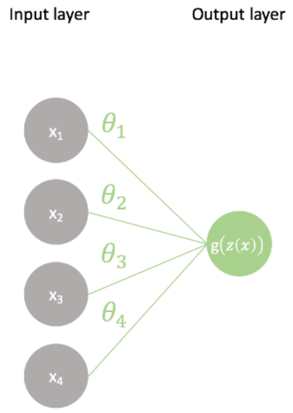
A computational model of a neuron

- A **neural network** is **defined** as a computing system that consist of a number of simple but highly interconnected elements or nodes, called 'neurons', which are organized in layers which process information using dynamic state responses to external inputs.

- Neural networks are a biologically-inspired algorithm that attempt to mimic the functions of neurons in the brain. Each neuron acts as a computational unit, accepting input from the dendrites and outputting signal through the axon terminals. Actions are triggered when a specific combination of neurons are activated.

In logistic regression, we composed a linear model z(x)z(x) with the logistic function g(z)g(z) to form our predictor. This linear model was a combination of feature inputs x_ix_i and weights w_iw_i.

$$z(x)=w_1x_{1+}w_2x_2+ w_3x_{3+}w_4x_{4=}w^Tx+b$$



Logistic Regression

$$z(x) = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b$$

$$g(z) = \frac{1}{1+e^{-z}}$$

Video Content / Details of website for further learning (if any):
https://youtu.be/EYeF2e2IKEo

https://www.codementor.io/@james_aka_yale/a-gentle-introduction-to-neural-networks-for-machine-learning-hkijvz7lp

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 81-82

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 2

CSE

IV/VII-B

Course Name with Code: Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : III

Date of Lecture:

Topic of Lecture: Appropriate problems for Neural Network Learning

Introduction: (Maximum 5 sentences): An **artificial neural network** (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Artificial neural network

Neural networkconcept

Appropriate Problems For Neural Network Learning

Neural networks, as the name suggests, are modeled on neurons in the brain. They use artificial intelligence to untangle and break down extremely complex relationships. What sets neural networks apart from other machine-learning algorithms is that they make use of an architecture inspired by the neurons in the brain

- ➢ Instances are represented by many attribute-value pairs.
- ➢ The target function output may be discrete-valued, real-valued, or a vector of several real-valued or discrete-valued attributes.
- ➢ The training examples may contain errors.
- ➢ Long training times are acceptable.
- ➢ Fast evaluation of the learned target function may be required.
- ➢ The ability of humans to understand the learned target function is not important.

Characteristics of Artificial Neural Networks

An Artificial Neural Network consists of large number of "neuron" like processing elements.
All these processing elements have a large number of weighted connections between them.
The connections between the elements provide a distributed representation of data.
.

Types of Neural Networks

- Feedforward Neural Network – Artificial Neuron. ...
- Radial Basis Function Neural Network. ...
- Multilayer Perceptron. ...
- Convolutional Neural Network. ...
- Recurrent Neural Network(RNN) – Long Short Term Memory. ...
- Modular Neural Network. ...
- Sequence-To-Sequence Models.

Neural Networks Applications :

- Text Classification,
- Information Extraction,
- Semantic Parsing,
- Question Answering,
- Paraphrase Detection,
- Language Generation,
- Multi-Document Summarization,
- Machine Translation,
- Speech And Character Recognition.

Video Content / Details of website for further learning (if any):

https://youtu.be/EYeF2e2IKEo

https://www.codementor.io/@james_aka_yale/a-gentle-introduction-to-neural-networks-for-machine-learning-hkijvz7lp

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 83-86

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

LECTURE HANDOUTS

L - 3

CSE

IV/VII-B

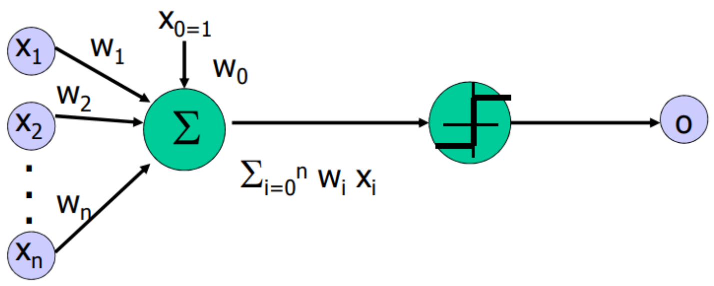
Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **III** **Date of Lecture:**

Topic of Lecture: Perceptron : Representational power of Perceptron
Introduction: (Maximum 5 sentences): A single neuron transforms given input into some output. Depending on the given input and weights assigned to each input, decide whether the neuron fired or not. Let's assume the neuron has 3 inputconnections and one output.
Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics) Artificial neural network Perceptron

Perceptron:



$$o(x_0,\ldots,x_n) = \begin{cases} 1 & \text{if } \Sigma_{i=0}^{n} w_i x_i > 0 \\ -1 & \text{otherwise} \end{cases}$$

- Perceptron is a Linear Threshold Unit (LTU).
- A perceptron takes a vector of real-valued inputs, calculates a linear combination of these inputs, then outputs 1 if the result is greater than some threshold and -1 otherwise.
- Given inputs xl through x_n , the output $o(x_1 , . . . , x_n)$ computed by the perceptron is:

$$o(x_1,\ldots,x_n) = \begin{cases} 1 \text{ if } w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n > 0 \\ -1 \text{ otherwise} \end{cases}$$

each w_i is a real-valued constant, or weight, that determines the contribution of input x_i to the perceptron output.

.

Perceptron – Learning

• Learning a perceptron involves choosing values for weights w_0 , …,w_n .

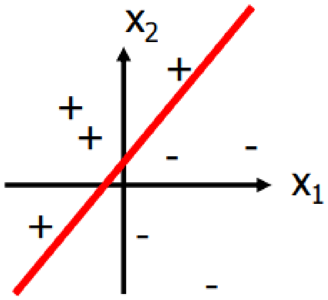
• The space H of candidate hypotheses considered in perceptron learning is the set of all possible real-valued weight vectors

$$H = \{\vec{w} \mid \vec{w} \in \Re^{(n+1)}\}$$

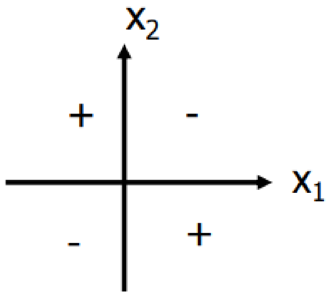
Representational Power of Perceptron :

- A perceptron represents a hyperplane decision surface in the n-dimensional space of instances.
- The perceptron outputs 1 for instances lying on one side of the hyperplane and outputs -1 for instances lying on the other side.
- The equation for this decision hyperplane is = 0
- Some sets of positive and negative examples cannot be separated by any hyperplane. Those that can be separated are called linearly separable sets of examples.
- A single perceptron can be used to represent many boolean functions.

– AND, OR, NAND, NOR are representable by a perceptron
– XOR cannot be representable by a perceptron.



Representable by a perceptron



NOT representable by a perceptron

Video Content / Details of website for further learning (if any):
https://youtu.be/x3joYu5VI38
https://www.simplilearn.com/what-is-perceptron-tutorial

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 86-87

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 4

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **III** **Date of Lecture:**

Topic of Lecture: Training rule- Gradient descent and Delta rule

Introduction: (Maximum 5 sentences): **Gradient descent** is an optimization algorithm used to minimize some function by iteratively moving in the direction of **steepest descent** as **defined** by the negative of the **gradient**.**machine learning**, we use **gradient descent** to update the parameters of our model.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Artificial neural network

Perceptron

Training Rule:

• To learn an acceptable weight vector is to begin with random weights, then iteratively apply the perceptron to each training example, modifying the perceptron weights whenever it misclassifies an example.

– If the training example classifies correctly, weights are not updated.

• This process is repeated, iterating through the training examples as many times as needed until the perceptron classifies all training examples correctly.

– Each pass through all of the training examples is called one epoch

• Weights are modified at each step according to perceptron training rule

$w_i = w_i + \Delta w_i$

$\Delta w_i = \eta \, (t - o)$

x_i t is the target value o is the perceptron output η is a small constant called learning rate

- If the output is correct (t=o) the weights w_i are not changed

- If the output is incorrect ($t \neq o$) the weights w_i are changed such that the output of the perceptron for the new weights is closer to t.

Gradient Descent and the Delta Rule:

- **Gradient descent** is used in supervised machine learning. Supervised machine learning dataset has true labels.
- Machine learning model predicts labels during training. Machine learning model has error equal to the difference between true label and predicted label for each sample of dataset.
- The perceptron rule finds a successful weight vector when the training examples are linearly

separable, it can fail to converge if the examples are not linearly separable.

- The delta rule overcomes this difficulty.
- If the training examples are not linearly separable, the delta rule converges toward a best-fit approximation to the target concept.
- The key idea behind the delta rule is to use gradient descent to search the hypothesis space of possible weight vectors to find the weights that best fit the training examples.
- The delta rule is important because gradient descent provides the basis for the **Backpropagation** Algorithm, which can learn networks with many interconnected units.

Gradient:

$$\nabla E[w0,\ldots,wn] = [\partial E/\partial w0,\ldots\ \partial E/\partial wn]$$

Training Rule for Gradient Descent

- For each weight wi
  - **wi = wi + Δwi**
  - **Δwi = - η** ∇E[wi]

η is a small constant called learning rate.

$$\Delta wi = -\ \eta\ \nabla E[wi]$$
$$= -\ \eta\ (\ \partial E/\partial wi\ )$$

Comparison Perceptron And Gradient Descent Rule

Perceptron learning rule guaranteed to succeed if

- Training examples are linearly separable
- Sufficiently small learning rate η

Linear unit training rules uses gradient descent

- Guaranteed to converge to hypothesis with minimum squared error
- Given sufficiently small learning rate η
- Even when training data contains noise
- Even when training data not separable by H

Video Content / Details of website for further learning (if any):

https://youtu.be/_x2-XzOovf0

https://towardsdatascience.com/gradient-descent-3a7db7520711

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 88-94

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 5

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : III

Date of Lecture:

Topic of Lecture: Multi layer networks and Backpropagation algorithm

Introduction: (Maximum 5 sentences): A fully connected **multi-layer** neural **network** is called a **Multilayer Perceptron** (MLP). It has 3 layers including one hidden layer. If it has more than 1 hidden layer, it is called a **deep** ANN. An MLP is a typical example of a feedforward artificial neural **network**.

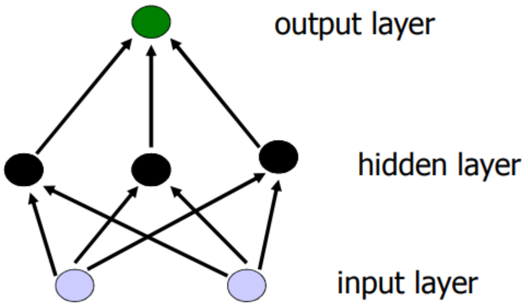
Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

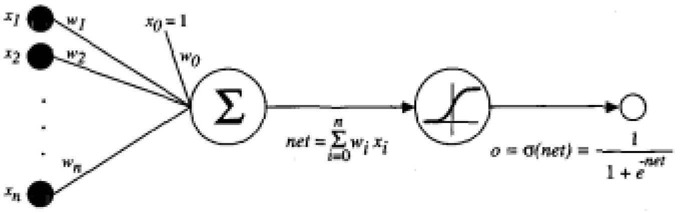
Artificial neural network

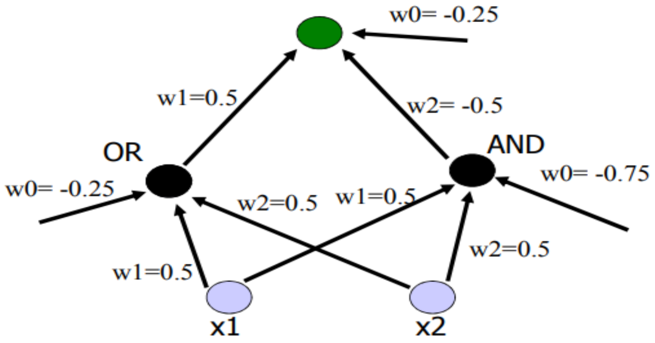
Perceptron

Multi-Layer Networks

- Single perceptron can only express linear decision surfaces.
- Multilayer networks are capable of expressing a rich variety of nonlinear decision surfaces.




x_1
x_2
x_n
w_1
w_2
w_n
$x_0 = 1$
w_0
Σ
$net = \sum_{i=0}^{n} w_i x_i$
$o = \sigma(net) = \frac{1}{1+e^{-net}}$



OR: $0.5*x1 + 0.5*x2 - 0.25 > 0$

AND: $0.5*x1 + 0.5*x2 - 0.75 > 0$

XOR: $0.5*x1 - 0.5*x2 - 0.25 > 0$

backpropagation algorithm is a method for training the weights in a multilayer feed-forward neural network. As such, it requires a network structure to be defined of one or more layers where one layer is fully connected to the next layer.

Algorithm:

- Create a feed-forward network with n_i inputs, n_{hidden} hidden units, and n_{out} output units.
- Initialize each w_i to some small random value (e.g., between -.05 and .05).
- Until the termination condition is met, Do

  – For each training example , Do

  // Propagate the input forward through the network:

  1. Input the instance (x_1 ,…,x_n) to the network and compute the network outputs o_k for every unit

  // Propagate the errors backward through the network:

  2. For each output unit k, calculate its error term δ_k

     $$\delta_k = o_k(1-o_k)(t_k-o_k)$$

  3. For each hidden unit h, calculate its error term δ_h

     $$\delta_h=o_h(1-o_h)\ \Sigma_k\ w_{h,k}\ \delta_k$$

  4. For each network weight $w_{i,j}$, Do

     $$w_{i,j}=w_{i,j}+\Delta w_{i,j}$$

     Where

     $$\Delta w_{i,j}= \eta\ \delta_j\ x_{i,j}$$

Video Content / Details of website for further learning (if any):

https://youtu.be/u5GAVdLQyIg

https://machinelearningmastery.com/neural-networks-crash-course/

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 95-99

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

LECTURE HANDOUTS

L - 6

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **III** **Date of Lecture:**

Topic of Lecture: Derivation & Remarks of the backpropagation rule

Introduction: (Maximum 5 sentences): The Backpropagation algorithm is used to learn the weights of a multilayer neural network with a fixed architecture. It performs gradient descent to try to minimize the sum squared error between the network's output values and the given target values.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Perceptron.
Backpropagation Algorithm

Derivation Of The Backpropagation Rule

- E_d is the error on training example d, summed over all output units in the network

$$E_d(\vec{w}) \equiv \frac{1}{2}\sum_{k\in outputs}(t_k - o_k)^2$$

- Here outputs is the set of output units in the network, tk is the target value of unit k for training example d, and ok is the output of unit k given training example d. The derivation of the stochastic gradient descent rule is conceptually straightforward, but requires keeping track of a number of subscripts and variables.
- x_{ji} = the ith input to unit j
- w_{ji} = the weight associated with the ith input to unit j
- $net_j = \sum_i wij\, x_{ji}$ (the weighted sum of inputs for unit j)
- o_j = the output computed by unit j
- t = the target output for unit j
- a = the sigmoid function
- outputs = the set of units in the final layer of the network
- Downstream(j) = the set of units whose immediate inputs include the output of unit j

Case 1: Training Rule for Output Unit Weights.

$$\Delta w_{ji} = -\eta\frac{\partial E_d}{\partial w_{ji}} = \eta\ (t_j - o_j)\ o_j(1-o_j)x_{ji}$$

Case 2: Training Rule for Hidden Unit Weights

$$\delta_j = o_j(1 - o_j) \sum_{k \in Downstream(j)} \delta_k \, w_{kj}$$

Remarks On The Backpropagation Algorithm

- Convergence and Local Minima
- Representational Power of Feedforward Networks
  - Boolean functions.
  - Continuous functions.
  - Arbitrary functions.
- Hypothesis Space Search and Inductive Bias
- Hidden Layer Representations
- Generalization, Overfitting, and Stopping Criterion

Video Content / Details of website for further learning (if any):

https://youtu.be/0e0z28wAWfg
https://www.cse.unsw.edu.au/~cs9417ml/MLP2/BackPropagation.html

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 101-111

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 7

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **III** **Date of Lecture:**

Topic of Lecture: Advanced topics in Neural Networks

Introduction: (Maximum 5 sentences): gradient descent can be performed for any function E that is differentiable with respect to the parameterized hypothesis space.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Artificial neural network

Backpropagation Algorithm

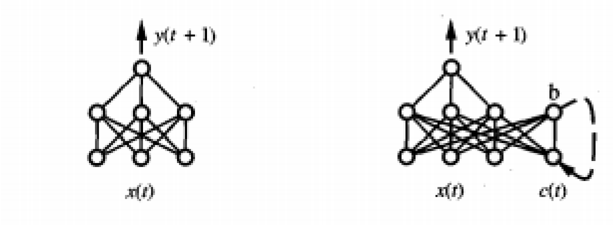
Alternative Error Minimization Procedures

- One optimization method, known as line search, involves a different approach to choosing the distance for the weight update. In particular, once a line is chosen that specifies the direction of the update, the update distance is chosen by finding the minimum of the error function along this line. Notice this can result in a very large or very small weight update, depending on the position of the point along the line that minimizes error.

- alternative error-minimization methods sometimes lead to improved efficiency in training the network, methods such as conjugate gradient tend to have no significant impact on the generalization error of the final network.

Recurrent Networks

- network topologies that correspond to acyclic directed graphs. Recurrent networks are artificial neural networks that apply to time series data and that use outputs of network units at time t as the input to other units at time t + 1. In this way, they support a form of directed cycles in the network. To illustrate, consider the time series prediction task of predicting the next day's stock market average y(t + 1) based on the current day's economic indicators x(t). Given a time series of such data, one obvious approach is to train a feedforward network to predict y(t + 1) as its output, based on the input values x(t) .



(a) Feedforward network (b) Recurrent network

Dynamically Modifying Network Structure

- A variety of methods have been proposed to dynamically grow or shrink the number of network units and interconnections in an attempt to improve generalization accuracy and training efficiency.

- A second idea for dynamically altering network structure is to take the opposite approach. Instead of beginning with the simplest possible network and adding complexity, we begin with a complex network and prune it as we find that certain connections are inessential.

- In general, techniques for dynamically modifying network structure have met with mixed success. It remains to be seen whether they can reliably improve on the generalization accuracy of backpropagation. However, they have been shown in some cases to provide significant improvements in training times.

Video Content / Details of website for further learning (if any):
https://youtu.be/z_54eqHPUfM
https://towardsdatascience.com/advanced-topics-in-neural-networks-f27fbcc638ae

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 117-119

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 8

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: III** **Date of Lecture:**

Topic of Lecture: Alternative error function

Introduction: (Maximum 5 sentences): gradient descent can be performed for any function E that is differentiable with respect to the parameterized hypothesis space.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Artificial neural network
Backpropagation Algorithm

Alternative error function

- Backpropagation algorithm defines E in terms of the sum of squared errors of the network, other definitions have been suggested in order to incorporate other constraints into the weight-tuning rule. For each new definition of E a new weight-tuning rule for gradient descent must be derived.

- Adding a penalty term for weight magnitude. As discussed above, we can add a term to E that increases with the magnitude of the weight vector. This causes the gradient descent search to seek weight vectors with small magnitudes, thereby reducing the risk of overfitting

$$E(\vec{w}) \equiv \frac{1}{2}\sum_{d\in D}\sum_{k\in outputs}(t_{kd}-o_{kd})^2 + \gamma\sum_{i,j} w_{ji}^2$$

- Adding a term for errors in the slope, or derivative of the target function. In some cases, training information may be available regarding desired derivatives of the target function, as well as desired values

$$E(\vec{w}) \equiv \frac{1}{2}\sum_{d\in D}\sum_{k\in outputs}\left[(t_{kd}-o_{kd})^2 + \mu\sum_{j\in inputs}\left(\frac{\partial t_{kd}}{\partial x_d^j} - \frac{\partial o_{kd}}{\partial x_d^j}\right)^2\right]$$

- Minimizing the cross entropy of the network with respect to the target values. Consider learning a probabilistic function, such as predicting whether a loan applicant will pay back a loan based on attributes such as the applicant's age and bank balance

$$\sum_{d \in D} t_d \log o_d + (1 - t_d) \log(1 - o_d)$$

- o_d is the probability estimate output by the network for training example d, and td is the 1 or 0 target value for training example d. when and why the most probable network hypothesis is the one that minimizes this cross entropy and derives the corresponding gradient descent weight-tuning rule for sigmoid units. That chapter also describes other conditions under which the most probable hypothesis is the one that minimizes the sum of squared errors.

Video Content / Details of website for further learning (if any):
https://youtu.be/GJXKOrqZauk
https://www.guru99.com/backpropogation-neural-network.html

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 117-119

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 9

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : III

Date of Lecture:

Topic of Lecture: Error minimization procedures & Recurrent networks-Dynamically modified Network structure.

Introduction: (Maximum 5 sentences): gradient descent can be performed for any function E that is differentiable with respect to the parameterized hypothesis space.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Artificial neural network
Backpropagation Algorithm

Error minimization procedures

- gradient descent is one of the most general search methods for finding a hypothesis to minimize the error function, it is not always the most efficient. It is not uncommon for backpropagation to require tens of thousands of iterations through the weight update loop when training complex networks

- One optimization method, known as line search, involves a different approach to choosing the distance for the weight update. In particular, once a line is chosen that specifies the direction of the update, the update distance is chosen by finding the minimum of the error function along this line.

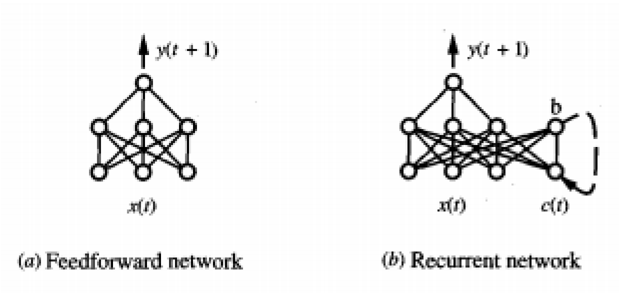
- A second method, that builds on the idea of line search, is called the conjugate gradient method. Here, a sequence of line searches is performed to search for a minimum in the error surface.

- alternative error-minimization methods sometimes lead to improved efficiency in training the network, methods such as conjugate gradient tend to have no significant impact on the generalization error of the final network. The only likely impact on the final error is that different error-minimization procedures may fall into different local minima

Recurrent networks

- Recurrent networks are artificial neural networks that apply to time series data and that use outputs of network units at time t as the input to other units at time t + 1. In this way, they support a form of directed cycles in the network. To illustrate, consider the time series prediction task of predicting the next day's stock market average y(t + 1) based on the current day's economic indicators x(t). Given a time series of such data, one obvious approach is to train a feedforward network to predict y(t + 1) as its output, based on the input values x(t)

- One limitation of such a network is that the prediction of y(t + 1) depends only on x(t) and cannot capture possible dependencies of y (t + 1) on earlier values of x.



(a) Feedforward network (b) Recurrent network

Dynamically modified Network structure.

- A variety of methods have been proposed to dynamically grow or shrink the number of network units and interconnections in an attempt to improve generalization accuracy and training efficiency.

- A second idea for dynamically altering network structure is to take the opposite approach. Instead of beginning with the simplest possible network and adding complexity, we begin with a complex network and prune it as we find that certain connections are inessential.

- general, techniques for dynamically modifying network structure have met with mixed success. It remains to be seen whether they can reliably improve on the generalization accuracy of backpropagation. However, they have been shown in some cases to provide significant improvements in training times.

Video Content / Details of website for further learning (if any):
https://youtu.be/GJXKOrqZauk
https://www.guru99.com/backpropogation-neural-network.html

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 118-119

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

DESIGNING YOUR FUTURE
Estd. 2000

IQAC

LECTURE HANDOUTS

L - 1

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : IV **Date of Lecture:**

Topic of Lecture: Introduction-KNN learning

Introduction: (Maximum 5 sentences): Instance-based learning methods such as nearest neighbor and locally weighted regression are conceptually straightforward approaches to approximating real-valued or discrete-valued target functions. Learning in these algorithms consists of simply storing the presented training data.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
KNN Learning concepts
KNN problems

K Nearest Neighbor Algorithm

- When a new query instance is encountered, a set of similar related instances is retrieved from memory and used to classify the new query instance.

- The most basic instance-based method is the k-nearest neighbor algorithm. The nearest neighbors of an instance are defined in terms of the standard Euclidean distance.

- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

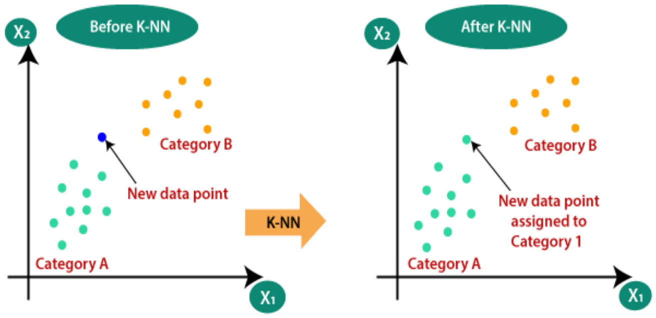
$$\langle a_1(x), a_2(x), \ldots a_n(x) \rangle$$

- where $a_r(x)$ denotes the value of the r^{th} attribute of instance x. Then the distance between two instances x_i and $x_j$is defined to be $d(x_i, x_j)$, where .

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2}$$

Example

- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
- **Step-6:** Our model is ready.



- For every training example, the polyhedron indicates the set of query points whose classification will be completely determined by that training example.
- Query points outside the polyhedron are closer to some other training example.

Video Content / Details of website for further learning (if any):
https://youtu.be/4HKqjENq9OU

https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 230-231

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 2

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : IV Date of Lecture:

Topic of Lecture: Problems on KNN

Introduction: (Maximum 5 sentences): Instance-based learning methods such as nearest neighbor and locally weighted regression are conceptually straightforward approaches to approximating real-valued or discrete-valued target functions. Learning in these algorithms consists of simply storing the presented training data.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

KNN Learning concepts
KNN problems

Problems on KNN

Advantages of KNN Algorithm:

- It is simple to implement.
- It is robust to the noisy training data
- It can be more effective if the training data is large.

Disadvantages of KNN Algorithm:

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples.

Remarks on KNN

- The distance-weighted k-nearest neighbor algorithm is a highly effective inductive inference method for many practical problems. It is robust to noisy training data and quite effective when it is provided a sufficiently large set of training data.

- One **practical issue** in applying k-nearest neighbor algorithms is that **the distance between instances is calculated based on all attributes of the instance**

- This lies in contrast to methods such as rule and decision tree learning systems that select only a subset of the instance attributes when forming the hypothesis.To see the effect of this policy, consider applying k-nearest neighbor to a problem in which **each instance is described by 20 attributes**, but where **only 2 of these attributes are relevant to determining the classification** for the particular target function.

- In this case, instances that have identical values for the 2 relevant attributes may nevertheless be distant from one another in the 20-dimensional instance space. As a result, the **similarity metric used by k-nearest neighbor--depending on all 20 attributes-will be misleading**.

- The **distance between neighbors will be dominated by the large number of irrelevant attributes**. This difficulty, which **arises** when **many irrelevant attributes are present**, is sometimes referred to as the **curse of dimensionality**.

- Nearest-neighbor approaches are especially sensitive to this problem.One interesting approach to **overcoming this problem** is to **weight each attribute differently** when calculating the distance between two instances

- One **additional practical issue** in applying k-nearest neighbor is **efficient memory indexing**. Because this algorithm **delays all processing** until a new query is received, significant computation can be required to process each new query.

Video Content / Details of website for further learning (if any):
https://youtu.be/4HKqjENq9OU

https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_knn_algorithm_finding_nearest_neighbors.html

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 231-234

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 3

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :Dr.N.Naveenkumar

Unit : IV **Date of Lecture:**

Topic of Lecture: Distance- Weighted Nearest Neighbor algorithm

Introduction: (Maximum 5 sentences): One obvious refinement to the k-nearest neighbor algorithm is to weight the contribution of each of the k neighbors according to their distance to the query point x$_q$, giving greater weight to closer neighbors.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

KNN Learning concepts

KNN problems

Distance- Weighted Nearest Neighbor algorithm

- This can be accomplished by replacing the final line of the algorithm by

$$\hat{f}(x_q) \leftarrow \underset{v \in V}{\operatorname{argmax}} \sum_{i=1}^{k} w_i \delta(v, f(x_i))$$

Where

$$w_i \equiv \frac{1}{d(x_q, x_i)^2}$$

- We can distance-weight the instances for real-valued target functions in a similar fashion, replacing the final line of the algorithm in this case by

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i}$$

- The only disadvantage of considering all examples is that our classifier will run more slowly.
- If all training examples are considered when classifying a new query instance, we call the algorithm a global method.
- If only the nearest training examples are considered, we call it a local method.

Video Content / Details of website for further learning (if any):
https://youtu.be/JCUtq5T0DOY

https://www.geeksforgeeks.org/weighted-k-nn/

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 233-236

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 4

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **IV**

Date of Lecture:

Topic of Lecture: Locally weighted regression

Introduction: (Maximum 5 sentences): Locally weighted regression uses nearby or distance-weighted training examples to form this local approximation to f. The nearest-neighbor approaches described in the previous section can be thought of as approximating the target function f (x) at the single query point x = x_q.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
KNN Learning concepts
KNN problems

Locally weighted regression

- Locally weighted regression is a generalization of this approach.
- It constructs an explicit approximation to f over a local region surrounding x_q.
- Locally weighted regression uses nearby or distance-weighted training examples to form this local approximation to f.
- The phrase "locally weighted regression" is called local because the function is approximated based a only on data near the query point, weighted because the contribution of each training example is weighted by its distance from the query point, and regression because this is the term used widely in the statistical learning community for the problem of approximating real-valued functions.
- Let us consider the case of locally weighted regression in which the target function f is approximated near $x_q$using a linear function of the form.

$$\hat{f}(x) = w_0 + w_1 a_1(x) + \cdots + w_n a_n(x)$$

Therefore, we derived methods to choose weights that minimize the squared error summed over the set D of training examples

$$E \equiv \frac{1}{2} \sum_{x \in D} (f(x) - \hat{f}(x))^2$$

which led us to the gradient descent training rule

$$\Delta w_j = \eta \sum_{x \in D} (f(x) - \hat{f}(x)) a_j(x)$$

Three possible criteria are given below. Note we write the error $E(x_q)$ to emphasize the fact that now the error is being defined as a function of the query point x_q.

1. Minimize the squared error over just the k nearest neighbors

$$E_1(x_q) \equiv \frac{1}{2} \sum_{x \in \, k \text{ nearest nbrs of } x_q} (f(x) - \hat{f}(x))^2$$

2. Minimize the squared error over the entire set D of training examples, while weighting the error of each training example by some decreasing function K of its distance from x_q:

$$E_2(x_q) \equiv \frac{1}{2} \sum_{x \in D} (f(x) - \hat{f}(x))^2 \; K(d(x_q, x))$$

3. Combine 1 and 2:

$$E_3(x_q) \equiv \frac{1}{2} \sum_{x \in \, k \text{ nearest nbrs of } x_q} (f(x) - \hat{f}(x))^2 \; K(d(x_q, x))$$

If we choose criterion three above and rederive the gradient descent rule, we obtain the following training rule

$$\Delta w_j = \eta \sum_{x \in \, k \text{ nearest nbrs of } x_q} K(d(x_q, x)) \; (f(x) - \hat{f}(x)) \; a_j(x)$$

Video Content / Details of website for further learning (if any):
https://youtu.be/vcEuUEY7BtM

https://www.geeksforgeeks.org/ml-locally-weighted-linear-regression/

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 237-237.

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 5

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **IV** **Date of Lecture:**

Topic of Lecture: Remarks on locally weighted regression

Introduction: (Maximum 5 sentences): Locally weighted regression uses nearby or distance-weighted training examples to form this local approximation to f. The nearest-neighbor approaches described in the previous section can be thought of as approximating the target function f (x) at the single query point x = x_q.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Knn Algorithm
locally weighted regression

Locally weighted regression

- Locally weighted regression is a generalization of this approach.
- It constructs an explicit approximation to f over a local region surrounding x_q.
- Locally weighted regression uses nearby or distance-weighted training examples to form this local approximation to f.
- The phrase "locally weighted regression" is called local because the function is approximated based a only on data near the query point, weighted because the contribution of each training example is weighted by its distance from the query point, and regression because this is the term used widely in the statistical learning community for the problem of approximating real-valued functions.

Remarks on locally weighted regression

- The literature on locally weighted regression contains a broad range of alternative methods for distance weighting the training examples

- a range of methods for locally approximating the target function.

- In most cases, the target function is approximated by a constant, linear, or quadratic function.

- More complex functional forms are not often found because.

- The cost of fitting more complex functions for each query instance is prohibitively high, and
    1. These simple approximations model the target function quite well over a sufficiently small subregion of the instance space.

Video Content / Details of website for further learning (if any):

https://youtu.be/vcEuUEY7BtM

https://www.geeksforgeeks.org/ml-locally-weighted-linear-regression/

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 237-238.

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 6

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: IV** **Date of Lecture:**

Topic of Lecture: Radial basis functions

Introduction: (Maximum 5 sentences): One approach to function approximation that is closely related to distance-weighted regression and also to artificial neural networks is learning with radial basis functions.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Knn Algorithm
locally weighted regression

Radial basis functions

- In this approach, the learned hypothesis is a function of the form

$$\hat{f}(x) = w_0 + \sum_{u=1}^{k} w_u K_u(d(x_u, x))$$

Where

Each x_u is an instance from X

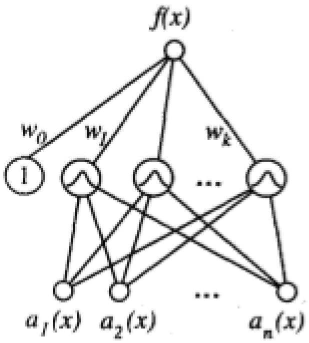
The kernel function K_u $(d(x_u, x))$ is defined so that it decreases as the distance $d(x_u, x)$ increases.

k is a user- provided constant that specifies the number of kernel functions to be included.

$$K_u(d(x_u, x)) = e^{\frac{1}{2\sigma_u^2} d^2(x_u, x)}$$

- Given a set of training examples of the target function, RBF networks are typically trained in a two-stage process.
- First, the number k of hidden units is determined and each hidden unit u is defined by choosing the values of x_u and σ_u^2: that define its kernel function $K_u(d(x_u, x))$.
- Second, the weights w, are trained to maximize the fit of the network to the training data, using the global error criterion given**.**

Example



- radial basis function networks provide a global approximation to the target function, represented by a linear combination of many local kernel functions.
- one key advantage to RBF networks is that they can be trained much more efficiently than feedforward networks trained with backpropagation.
- This follows from the fact that the input layer and the output layer of an RBF are trained separately.

Video Content / Details of website for further learning (if any):
https://youtu.be/OUtTI99uRf4

https://towardsdatascience.com/radial-basis-functions-neural-networks-all-we-need-to-know-9a88cc053448

Important Books/Journals for further learning including the page nos.: .

Tom michelle “ Machine Learning”, Page no : 238-240.

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

LECTURE HANDOUTS

L - 7

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: IV** **Date of Lecture:**

Topic of Lecture: Case –Based reasoning

Introduction: (Maximum 5 sentences):CBR has been applied to problems such as **conceptual design of mechanical devices based on a stored library of previous designs reasoning about new legal cases based on previous rulings** and solving planning and scheduling problems by reusing and combining portions of previous solutions to similar problems.

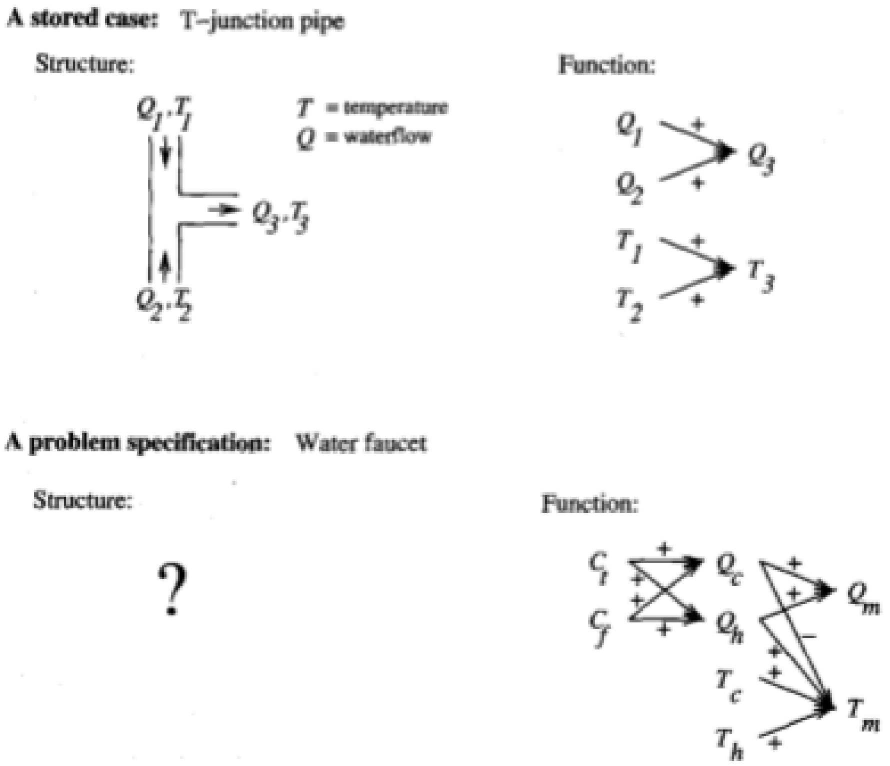
Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Locally weighted regression
Radial basis functions

Case –Based reasoning

- Instance-based methods such as k-nearest neighbour and locally weighted regression share three key properties.
  1. (1) First, they are lazy learning methods in that they defer the decision of how to generalize beyond the training data until a new query instance is observed.
  2. (2) Second, they classify new query instances by analyzing similar instances while ignoring instances that are very different from the query.
  3. (3) Third, they represent instances as real-valued points in an n-dimensional Euclidean space.
- Case-based reasoning (CBR) is a learning paradigm **based on the first two of these principles**, but **not the third**.
- In CBR, instances are typically represented using more rich symbolic descriptions, and the methods used to retrieve similar instances are correspondingly more elaborate.

Example

- The CADET system employs case- based reasoning to assist in the conceptual design of simple mechanical devices such as water faucets.
- It uses a library containing approximately 75 previous designs and design fragments to suggest conceptual designs to meet the specifications of new design problems.



- One current research issue in case-based reasoning is to develop improved methods for indexing cases.
- The central issue here is that syntactic similarity measures (e.g., subgraph isomorphism between function graphs) provide only an approximate indication of the relevance of a particular case to a particular problem.

Video Content / Details of website for further learning (if any):
https://youtu.be/8p7dzVSr4XU

https://artint.info/html/ArtInt_190.html

Important Books/Journals for further learning including the page nos:
MATERIAL

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L -8

CSE

IV/VII-B

Course Name with Code: Machine Learning Techniques -16CSE14

Course Teacher :**Dr.N.Naveenkumar**

Unit : **IV** **Date of Lecture:**

Topic of Lecture: Remarks on Lazy and Eager Learning

Introduction: (Maximum 5 sentences): Lazy methods may consider the query instance x, when deciding how to generalize beyond the training data D. Eager methods cannot. By the time they observe the query instance x, they have already chosen their (global) approximation to the target function.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Knn Algorithm
locally weighted regression

Remarks on Lazy and Eager Learning

- Three lazy learning methods:

  1. The k-nearest neighbor algorithm,
  2. locally weighted regression, and
  3. case-based reasoning

  .

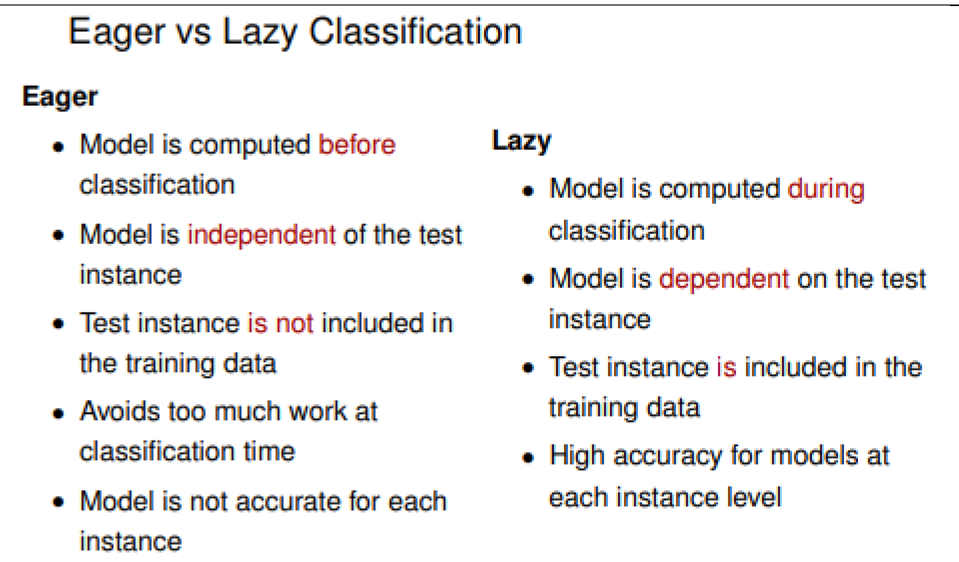
- We call these methods lazy because they **defer the decision of how to generalize beyond the training data until each new query instance is encountered**.

- One eager learning method:
- The method for learning radial basis function networks.
- We call this method eager because **it generalizes beyond the training data before observing the new query,** committing at training time to the network structure and weights that define its approximation to the target function.

Two kinds of differences:

(1) differences in computation time

(2) differences in the classifications produced for new queries



Eager vs Lazy Classification

Eager

- Model is computed before classification
- Model is independent of the test instance
- Test instance is not included in the training data
- Avoids too much work at classification time
- Model is not accurate for each instance

Lazy

- Model is computed during classification
- Model is dependent on the test instance
- Test instance is included in the training data
- High accuracy for models at each instance level

- To summarize lazy methods have the option of selecting a different hypothesis or local approximation to the target function for each query instance.
- Eager methods using the same hypothesis space are more restricted because they must commit to a single hypothesis that covers the entire instance space.

Video Content / Details of website for further learning (if any):
https://youtu.be/xk-AKHUCdGc

http://jmvidal.cse.sc.edu/talks/instancelearning/lazyandeagerlearning.html

Important Books/Journals for further learning including the page nos.: .

Tom michelle " Machine Learning", Page no : 244-247.

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 9

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **:Dr.N.Naveenkumar**

Unit **: IV** **Date of Lecture:**

Topic of Lecture: Deep Learning

Introduction: (Maximum 5 sentences): **Deep learning** is a subset of **machine learning** where artificial **neural networks**, algorithms inspired by the human brain, **learn** from large amounts of data. **Deep learning** allows machines to solve complex problems even when using a data set that is very diverse, unstructured and inter-connected.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Machine Learning
Artificial Intelligence

Deep learning

- Deep learning is a type of machine learning (ML) and artificial intelligence (AI) that **imitates the way humans gain certain types of knowledge**.
- Deep learning is an important element of data science, which includes **statistics and predictive modeling**.
- It is extremely beneficial to data scientists who are tasked with **collecting, analyzing and interpreting large amounts of data**;
- deep learning makes this **process faster and easier**.
- At its simplest, deep learning can be thought of as a way to automate predictive analytics. While traditional machine learning algorithms are linear, deep learning algorithms are stacked in a hierarchy of increasing complexity and abstraction.

Deep learning vs. machine learning

- Deep learning is a subset of machine learning that differentiates itself through the way it solves problems. Machine learning requires a domain expert to identify most applied features. On the other hand, deep learning learns features incrementally, thus eliminating the need for domain expertise. This makes deep learning algorithms take much longer to train than machine learning algorithms, which only need a few seconds to a few hours. However, the reverse is true during testing. Deep learning algorithms take much less time to run tests than machine learning algorithms, whose test time increases along with the size of the data.

Examples of Deep Learning

- Deep learning research is a driving force behind many of the technologies we use every day -- from the voice control functions found in our smart devices to self-driving cars.

- Both Netflix and Amazon use deep learning algorithms to suggest products and shows to watch, intelligent virtual assistants (Alexa, Bixby, Cortana, Google Assistant or Siri) use deep learning to understand speech and the language humans use when they interact with them. Other examples of deep learning include colorization of black and white Images, autonomous vehicles, translators, facial recognition, classification and medical disease diagnoses.

Benefits or advantages of Deep Learning:

➡Robustness to natural variations in the data is automatically learned.

➡The same neural network based approach can be applied to many different applications and data types.

➡The deep learning architecture is flexible to be adapted to new problems in the future.

Drawbacks or **disadvantages of Deep Learning**:

➡It requires very large amount of data in order to perform better than other techniques.

➡It is extremely expensive to train due to complex data models. Moreover deep learning requires expensive GPUs and hundreds of machines. This increases cost to the users..

➡It is not easy to comprehend output based on mere learning and requires classifiers to do so. Convolutional neural network based algorithms perform such tasks.

Video Content / Details of website for further learning (if any):
https://youtu.be/6M5VXKLf4D4

https://www.investopedia.com/terms/d/deep-learning.asp

Important Books/Journals for further learning including the page nos.: .

Material

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 1

BIV/VII

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : **Dr.N.Naveenkumar**

Unit : **VDate of Lecture:**

Topic of Lecture: Graphical Model

Introduction: (Maximum 5 sentences):Graphical models represent **the interaction between variables visually** and have the advantage that inference over a large number of variables can be decomposed into a set of local calculations involving a small number of variables making use of conditional independencies.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Probability
Graphical Model

Graphical Model

- Graphical models, also called **Bayesian networks, belief networks, or probabilistic networks**, are composed of nodes and arcs between the nodes.
- Each node corresponds to a random variable, X, and has a value corresponding to the probability of the random variable, P(X).
- If there is a directed arc from node X to node Y, this indicates that X has a direct influence on Y. This influence is specified by the conditional probability P(Y|X).
- The network is a directed acyclic graph (DAG); namely, there are no cycles.
- The nodes and the arcs between the nodes define the structure of the network, and the conditional probabilities are the parameters given the structure.

Independence

X and Y are independent events if

$$P(X,Y)=P(X)P(Y)$$

Conditional Independence

X and Y are conditionally independent events given a third event Z if

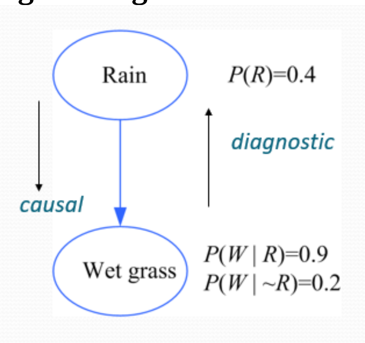
$$P(X,Y|Z)=P(X|Z)P(Y|Z)$$

which can also be rewritten as

$$P(X|Y,Z)=P(X|Z)$$

Example

---which models that rain causes the grass to get wet.



The joint is written as

$$P(R,W) = P(R)P(W|R)$$

calculate the individual (marginal) probability of wet grass by summing up over the possible values

$$\begin{aligned} P(W) &= \sum_R P(R,W) = P(W|R)P(R) + P(W|{\sim}R)P({\sim}R) \\ &= 0.9 \cdot 0.4 + 0.2 \cdot 0.6 = 0.48 \end{aligned}$$

knowing that the grass is wet, the probability that it rained can be calculated as follows:

$$P(R|W) = \frac{P(W|R)P(R)}{P(W)} = 0.75$$

Knowing that the grass is wet increased the probability of rain from 0.4 to 0.75; this is because P(W|R) is high and P(W|~R) is low.

Video Content / Details of website for further learning (if any):

https://youtu.be/2EqE9HDDRQc

https://medium.com/@jonathan_hui/machine-learning-graphical-model-b68b0c27a749

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 387-389

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 2

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V

Date of Lecture:

Topic of Lecture: Canonical cases for conditional Independence

Introduction: (Maximum 5 sentences): Graphical models represent **the interaction between variables visually** and have the advantage that inference over a large number of variables can be decomposed into a set of local calculations involving a small number of variables making use of conditional independencies.

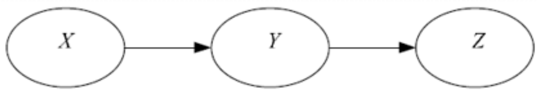
Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Probability

Graphical Model

Canonical Cases for Conditional Independence

Case 1: Head-to-tail Connection



(a) Model

- Three events may be connected serially.We see here that X and Z are independent given Y: Knowing Y tells Z everything; we write

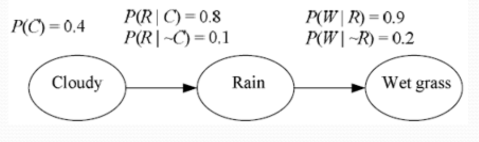
$$P(Z|Y,X)= P(Z|Y).$$

- We say that Y blocks the path from X to Z, or in other words, it separates them in the sense that if Y is removed, there is no path between X to Z. In this case, the joint is written as

$$P(X,Y,Z)=P(X)P(Y|X)P(Z|Y)$$

Writing the joint this way implies independence:

$$P(Z|X,Y) = \frac{P(X,Y,Z)}{P(X,Y)} = \frac{P(X)P(Y|X)P(Z|Y)}{P(X)P(Y|X)} = P(Z|Y)$$



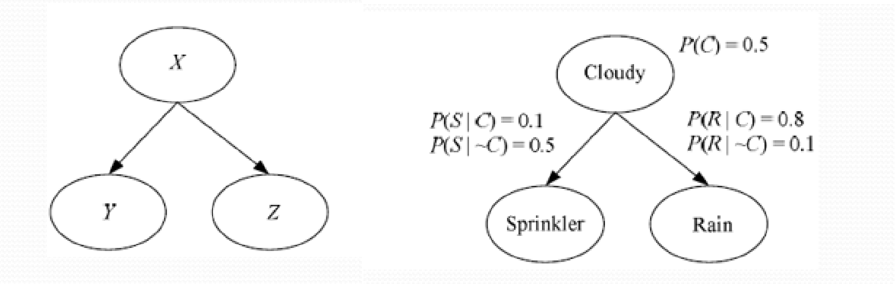
$$P(W|C) = P(W|R)P(R|C) + P(W|{\sim}R)P({\sim}R|C) = 0.76$$

$$P(C|W) = \frac{P(W|C)P(C)}{P(W)} = 0.65$$

Case 2: Tail-to-tail Connection

X may be the parent of two nodes Y and Z,

- *P*(*X*,*Y*,*Z*)=*P*(*X*)*P*(*Y*|*X*)*P*(*Z*|*X*)



Normally Y and Z are dependent through X; givenX, they become independent:

$$P(Y,Z|X) = \frac{P(X,Y,Z)}{P(X)} = \frac{P(X)P(Y|X)P(Z|X)}{P(X)} = P(Y|X)P(Z|X)$$

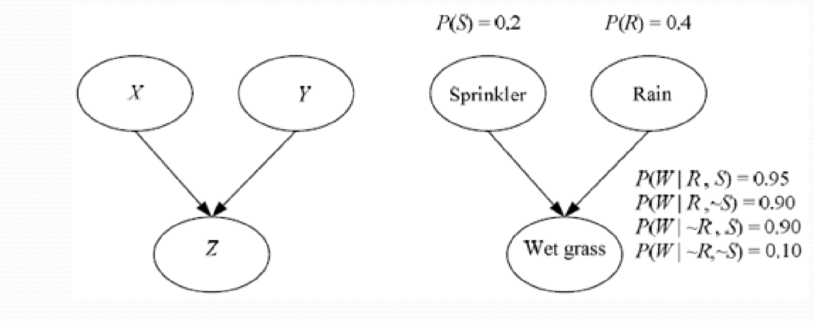
$$\begin{aligned} P(C|R) &= \frac{P(R|C)P(C)}{P(R)} = \frac{P(R|C)P(C)}{\sum_C P(R,C)} \\ &= \frac{P(R|C)P(C)}{P(R|C)P(C) + P(R|{\sim}C)P({\sim}C)} = 0.89 \end{aligned}$$

$$P(R|S) = \sum_C P(R,C|S) = P(R|C)P(C|S) + P(R|{\sim}C)P({\sim}C|S)$$

$$\begin{aligned} &= P(R|C)\frac{P(S|C)P(C)}{P(S)} + P(R|{\sim}C)\frac{P(S|{\sim}C)P({\sim}C)}{P({\sim}S)} \\ &= 0.22 \end{aligned}$$

Case 3: Head-to-head Connection

In a head-to-head node, there are two parents X and Y to a single node Z,



The joint density is written as

- *P*(*X*,*Y*,*Z*)=*P*(*X*)*P*(*Y*)*P*(*Z*|*X*,*Y*)

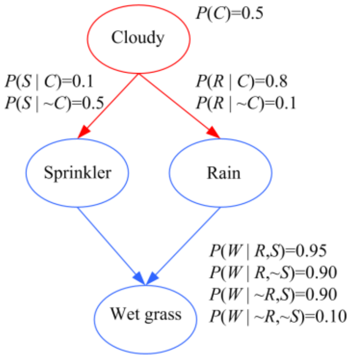
$$\begin{aligned} P(W) &= \sum_{R,S} P(W,R,S) \\ &= P(W|R,S)P(R,S) + P(W|{\sim}R,S)P({\sim}R,S) \\ &\quad + P(W|R,{\sim}S)P(R,{\sim}S) + P(W|{\sim}R,{\sim}S)P({\sim}R,{\sim}S) \\ &= P(W|R,S)P(R)P(S) + P(W|{\sim}R,S)P({\sim}R)P(S) \\ &\quad + P(W|R,{\sim}S)P(R)P({\sim}S) + P(W|{\sim}R,{\sim}S)P({\sim}R)P({\sim}S) \\ &= 0.52 \end{aligned}$$

$$\begin{aligned} P(W|S) &= \sum_R P(W,R|S) \\ &= P(W|R,S)P(R|S) + P(W|{\sim}R,S)P({\sim}R|S) \\ &= P(W|R,S)P(R) + P(W|{\sim}R,S)P({\sim}R) \\ &= 0.92 \end{aligned}$$

$$P(S|W) = \frac{P(W|S)P(S)}{P(W)} = 0.35$$

$$\begin{aligned} P(S|R,W) &= \frac{P(W|R,S)P(S|R)}{P(W|R)} = \frac{P(W|R,S)P(S)}{P(W|R)} \\ &= 0.21 \end{aligned}$$

Explaining away: Knowing that it has raineddecreases the probability that the sprinkler is on.



$$P(W|C) = \sum_{R,S} P(W,R,S|C)$$

We can calculate P(C|W)and have a diagnostic inference:

$$P(C|W) = \frac{P(W|C)P(C)}{P(W)}$$

As we have seen earlier, inference is also easier as the joint density is broken down into conditional densities of smaller groups of variables:

$$P(C,S,R,W) = P(C)P(S|C)P(R|C)P(W|S,R)$$

In the general case, when we have variables $X_1,...,X_d$, we write

$$P(X_1,\ldots X_d) = \prod_{i=1}^{d} P(X_i \,|\, \text{parents } (X_i))$$

Video Content / Details of website for further learning (if any):
https://youtu.be/2RQQTEy1ugs

https://towardsdatascience.com/conditional-independence-the-backbone-of-bayesian-networks-85710f1b35b

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 389-396

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

DESIGNING YOUR FUTURE
Estd. 2000

IQAC

LECTURE HANDOUTS

L - 3

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V

Date of Lecture:

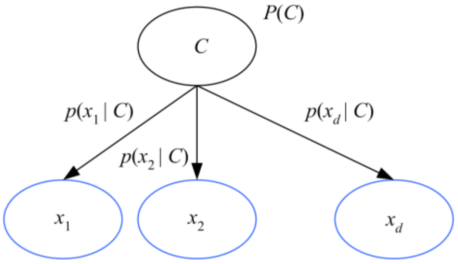
Topic of Lecture: Example Graphical Models

Introduction: (Maximum 5 sentences): Graphical models represent **the interaction between variables visually** and have the advantage that inference over a large number of variables can be decomposed into a set of local calculations involving a small number of variables making use of conditional independencies.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)

Canonical cases
Graphical Model
.

Example Graphical Models
Naive Bayes' Classifier



- If the inputs are independent, we have the graph ,which is called the naive Bayes' classifier, because it ignores possible dependencies, namely, correlations, among the inputs and reduces a multivariate problem to a group of univariate problems:

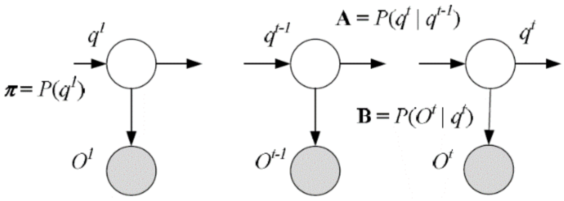
Given C, x_j are independent:

$$p(\mathbf{x}|C) = p(x_1|C)\, p(x_2|C) \ldots p(x_d|C)$$

Hidden Markov Model as a Graphical Model

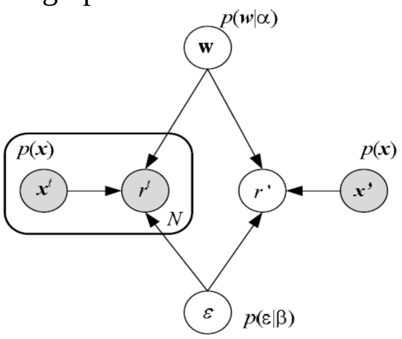
- Hidden Markov models (HMM), where three successive states q_{t-2}, q_{t-1}, q_t correspond to three states on a chain in a first-order Markov model. The state at time t, q_t, depends only on the state at time t−1, q_{t-1}, and given q_{t-1}, qt is independent of q_{t-2}

$$P(q_t|q_{t-1}, q_{t-2}) = P(q_t|q_{t-1})$$


q^1
$\pi = P(q^1)$
O^1
q^{t-1}
$\mathbf{A} = P(q^t \mid q^{t-1})$
q^t
$\mathbf{B} = P(O^t \mid q^t)$
O^{t-1}
O^t

Linear Regression

Linear regression can be visualized as a graphical model



$$p(r'|\mathbf{x}',\mathbf{r},\mathbf{X}) = \int p(r'|\mathbf{x}',\mathbf{w})p(\mathbf{w}\,|\,\mathbf{X},\mathbf{r})d\mathbf{w}$$

$$= \int p(r'|\mathbf{x}',\mathbf{w})\frac{p(\mathbf{r}\,|\,\mathbf{X},\mathbf{w})p(\mathbf{w})}{p(\mathbf{r})}d\mathbf{w}$$

$$\propto \int p(r'|\mathbf{x}',\mathbf{w})\prod_t p(r^t\,|\,\mathbf{x}^t,\mathbf{w})p(\mathbf{w})d\mathbf{w}$$

where the second line is due to Bayes' rule and the third line is due to the independence of instances in the training set.

Video Content / Details of website for further learning (if any):

https://youtu.be/c0AWH5UFyOk

https://www.coursera.org/learn/probabilistic-graphical-models

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no: 396-402

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L -4

CSE **IV/VII-B**

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V **Date of Lecture:**

Topic of Lecture: Combining Multiple Learners

Introduction: (Maximum 5 sentences): Learning algorithm dictates a certain model that comes with a set of assumptions. This inductive bias leads to error if the assumptions do not hold for the data. Learning is an ill-posed problem and with finite data, each algorithm converges to a different solution and fails under different circumstances

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Probability
Graphical Model

Combining Multiple Learners

Rationale

- There is no algorithm that induces the most accurate learner in any domain, all the time.
- Generate a group of base-learners which when combined has higher accuracy.
- By suitably combining multiple base- learners then, accuracy can be improved. Recently with computation and memory getting cheaper, such systems composed of multiple learners have become popular
- Different learners use different
- Algorithms: making different assumptions
- Hyperparameters: e.g number of hidden nodes in NN, k in k-NN
- Representations: diff. features, multiple sources of information
- Training sets: small variations in the sets or diff. subproblems

Combination Methods

- Static Structures
- Ensemble averaging (Sum,Product,Min rule)
- Bagging
- Boosting
- Error Correcting Output Codes
- Dynamic structures

- Mixture of Experts
- Hierarchical Mixture of Experts

Video Content / Details of website for further learning (if any):
https://youtu.be/LNrBcDfUhq0

https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 419-420.

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L -5

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V

Date of Lecture:

Topic of Lecture: Voting and Bagging

Introduction: (Maximum 5 sentences): The simplest way to combine multiple classifiers is by voting, which corresponds to taking a linear combination of the learners. Bagging is a voting method whereby base-learners are made different by training them over slightly different training sets.

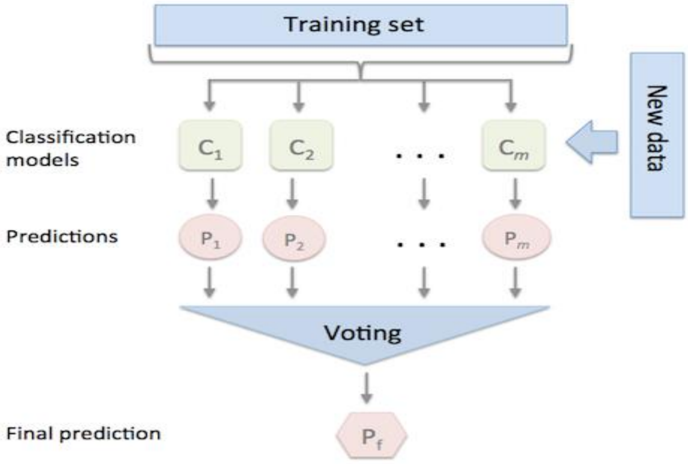
Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Graphical Model

Combining multiple learners

Voting and Bagging

- The `voting` is a meta-classifier for combining similar or conceptually different machine learning classifiers for classification via majority or plurality voting.
- It implements "hard" and "soft" voting



Hard Voting

- In hard voting, we predict the final class label as the class label that has been predicted most frequently by the classification models.

$$\hat{y} = mode\{C_1(\mathbf{x}), C_2(\mathbf{x}), \ldots, C_m(\mathbf{x})\}$$

- classifier 1 -> class 0
- classifier 2 -> class 0
- classifier 3 -> class 1

$$\hat{y} = mode\{0, 0, 1\} = 0$$

Continuing with the example from the previous section

- classifier 1 -> class 0
- classifier 2 -> class 0
- classifier 3 -> class 1

assigning the weights {0.2, 0.2, 0.6} would yield a prediction

$\hat{y} = 1$

$$\arg\max_i [0.2 \times i_0 + 0.2 \times i_0 + 0.6 \times i_1] = 1$$

Soft Voting

- In soft voting, we predict the class labels by averaging the class-probabilities (only recommended if the classifiers are well-calibrated).

$$\hat{y} = \arg\max_i \sum_{j=1}^{m} w_j p_{ij}$$

- where w_j is the weight that can be assigned to the j^{th} classifier.
- Assuming the example in the previous section was a binary classification task with class labels i∈{0,1}i∈{0,1}, our ensemble could make the following prediction:
  - $C_1(\mathbf{x}) \rightarrow [0.9, 0.1]$
  - $C_2(\mathbf{x}) \rightarrow [0.8, 0.2]$
  - $C_3(\mathbf{x}) \rightarrow [0.4, 0.6]$

Using uniform weights, we compute the average probabilities:

$$p(i_0 \mid \mathbf{x}) = \frac{0.9 + 0.8 + 0.4}{3} = 0.7$$

$$p(i_1 \mid \mathbf{x}) = \frac{0.1 + 0.2 + 0.6}{3} = 0.3$$

$$\hat{y} = \arg\max_i \left[p(i_0 \mid \mathbf{x}), p(i_1 \mid \mathbf{x})\right] = 0$$

However, assigning the weights {0.1, 0.1, 0.8} would yield a prediction

$\hat{y} = 1$:

$$p(i_0 \mid \mathbf{x}) = 0.1 \times 0.9 + 0.1 \times 0.8 + 0.8 \times 0.4 = 0.49$$

$$p(i_1 \mid \mathbf{x}) = 0.1 \times 0.1 + 0.2 \times 0.1 + 0.8 \times 0.6 = 0.51$$

$$\hat{y} = \arg\max_i \left[p(i_0 \mid \mathbf{x}), p(i_1 \mid \mathbf{x})\right] = 1$$

Bagging

- Bagging is a voting method whereby base-learners are made different by training them over slightly different training sets.

- Bootstrap Aggregation (or Bagging for short), is a simple and very powerful ensemble method. Bagging is the application of the Bootstrap procedure to a high-variance machine learning algorithm, typically decision trees.
- Bagging is used when the goal is to reduce the variance of a decision tree classifier.

Example: random forest

Bagging Steps:

- Suppose there are N observations and M features in training data set. A sample from training data set is taken randomly with replacement.
- A subset of M features are selected randomly and whichever feature gives the best split is used to split the node iteratively.
- The tree is grown to the largest.
- Above steps are repeated n times and prediction is given based on the aggregation of predictions from n number of trees.

Advantages:

- Reduces over-fitting of the model.
- Handles higher dimensionality data very well.
- Maintains accuracy for missing data.

Disadvantages:

- Since final prediction is based on the mean predictions from subset trees, it won't give precise values for the classification and regression model.

Video Content / Details of website for further learning (if any):

https://youtu.be/LNrBcDfUhq0

https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 424-430.

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 6

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V **Date of Lecture:**

Topic of Lecture: Boosting

Introduction: (Maximum 5 sentences):Boosting, we actively try to generate complementary base-learners by training the next learner on the mistakes of the previous learners. The original boosting algorithm combines three weak learners to generate a strong learner. A weak learner has error probability less than 1/2, which makes it better than random guessing on a two-class problem, and a strong learner has arbitrarily small error probability.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Graphical Model
Combining multiple learners

Boosting

Boosting is used to create a collection of predictors.

Example:Ada Boost

Boosting Steps:

- Draw a random subset of training samples d1 without replacement from the training set D to train a weak learner C1
- Draw second random training subset d2 without replacement from the training set and add 50 percent of the samples that were previously falsely classified/misclassified to train a weak learner C2
- Find the training samples d3 in the training set D on which C1 and C2 disagree to train a third weak learner C3
- Combine all the weak learners via majority voting.

Advantages

- Supports different loss function (we have used ‘binary:logistic’ for this example).

- Works well with interactions.

Disadvantages:

- Prone to over-fitting.
- Requires careful tuning of different hyper-parameters.

Video Content / Details of website for further learning (if any):

https://youtu.be/LNrBcDfUhq0

https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 424-430.

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 7

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V

Date of Lecture:

Topic of Lecture: Stacked generalization

Introduction: (Maximum 5 sentences):Stacked generalization is a general method of using a high-level model to combine lower- level models to achieve greater predictive accuracy.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Graphical Model
Combining multiple learners

Stacked generalization

- **Stacked generalization** is a way of combining multiple models that have been learned for a classification task.

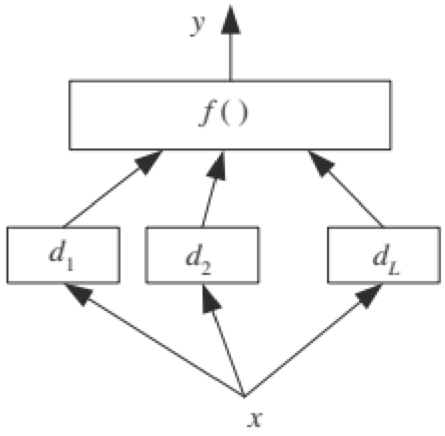
- Stacked generalization (or stacking) is a different way of combining multiple models, that introduces the concept of a meta learner. Although an attractive idea, it is less widely used than bagging and boosting. Unlike bagging and boosting, stacking may be (and normally is) used to combine models of different types. The procedure is as follows:

1. Split the training set into two disjoint sets.
2. Train several base learners on the first part.
3. Test the base learners on the second part.
4. Using the predictions from 3) as the inputs, and the correct responses as the outputs, train a higher level learner.

$$y = f(d_1, d_2, \ldots, d_L|\Phi)$$

- The combiner learns what the correct output is when the base-learners give a certain output combination.

- We cannot train the combiner function on the training data because the base-learners may be memorizing the training set; the combiner system should actually learn how the base learners make errors. Stacking is a means of estimating and correcting for the biases of the base-learners



1) to 3) are the same as cross-validation, but instead of using a winner-takes-all approach, we combine the base learners, possibly nonlinearly.

Video Content / Details of website for further learning (if any):
https://youtu.be/DCrcoh7cMHU
http://machine-learning.martinsewell.com/ensembles/stacking/

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, "Introduction to Machine Learning", Second Edition, MITPress,2013,Page no: 447-450.

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L - 8

CSE

IV/VII-B

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher **: Dr.N.Naveenkumar**

Unit **: V** **Date of Lecture:**

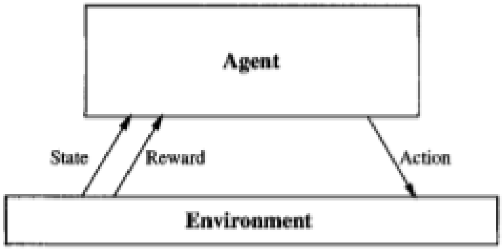
Topic of Lecture: Reinforcement Learning

Introduction: (Maximum 5 sentences):**Reinforcement learning** addresses the question of how an autonomous agent that senses and acts in its environment can learn to choose optimal actions to achieve its goals.

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)
Machine learning techniques
Machine learning Algorithm

Reinforcement learning

- The learning decision maker is called the agent. The agent interacts with the environment that includes everything outside the agent. The agent has sensors to decide on its state in the environment and takes an action that modifies its state. When the agent takes an action, the environment provides a reward.



$$s_0 \xrightarrow[r_0]{a_0} s_1 \xrightarrow[r_1]{a_1} s_2 \xrightarrow[r_2]{a_2} \dots$$

Goal: Learn to choose actions that maximize

$$r_0 + \gamma r_1 + \gamma^2 r_2 + \dots \text{ , where } 0 \leq \gamma < 1$$

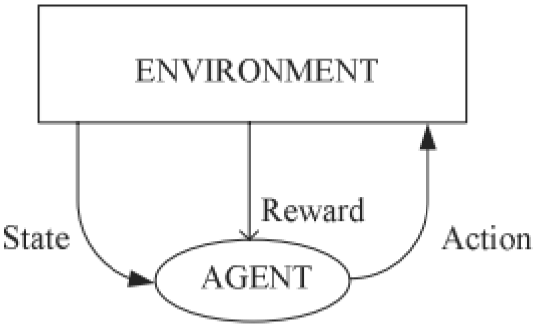
$$V^*(s_t) = \max_{\pi} V^{\pi}(s_t), \forall s_t$$

$$V^*(s_t) = \max_{a_t}\left(E[r_{t+1}] + \gamma \sum_{s_{t+1}} P(s_{t+1}|s_t, a_t) V^*(s_{t+1})\right)$$

$$\pi^*(s_t) : \text{Choose } a_t^* \text{ where } Q^*(s_t, a_t^*) = \max_{a_t} Q^*(s_t, a_t)$$

Model-Based Learning

- Once we have the optimal value function, the optimal policy is to choose the action that maximizes the value in the next state:



$$\pi^*(s_t) = \arg\max_{a_t}\left(E[r_{t+1}|s_t, a_t] + \gamma \sum_{s_{t+1} \in S} P(s_{t+1}|s_t, a_t) V^*(s_t + 1)\right)$$

- The agent interacts with an environment. At any state of the environment, the agent takes an action that changes the state and returns a reward.

Video Content / Details of website for further learning (if any):

https://youtu.be/LzaWrmKL1Z4

https://www.geeksforgeeks.org/what-is-reinforcement-learning/

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 450-454.

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu.

Estd. 2000

LECTURE HANDOUTS

L - 9

CSECSE

IV/VII-

Course Name with Code :Machine Learning Techniques -16CSE14
Course Teacher : Dr.N.Naveenkumar
Unit : V Date of Lecture:

Topic of Lecture: Learning task- Q learning-Example

Introduction: (Maximum 5 sentences):State and action, we may receive different rewards or move to different next states. What we do is keep a running average. This is known as the Q learning Q learning algorithm.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

Machine learning techniques
Reinforcement Learning

Learning task- Q learning-Example

- we formulate the problem of learning sequential control strategies more precisely.
- The task of the agent is to learn a policy,

$\pi : S \rightarrow A,$

for selecting its next action a, based on the current observed state s_t:that is

$\pi(s_t) = a_t.$

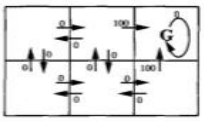
- One obvious approach is to require the policy that produces the greatest possible cumulative reward for the robot over time. To state this requirement more precisely, we define the cumulative value $V^{\pi}(s,)$ achieved by following an arbitrary policy π from an arbitrary initial

$$V^{\pi}(s_t) \equiv r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots$$
$$\equiv \sum_{i=0}^{\infty} \gamma^i r_{t+i}$$

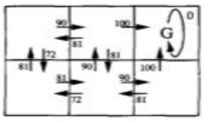
state s_t as follows:

We require that the agent learn a policy π that maximizes $V^{\pi}(s)$ for all states s. We will call such a policy an optimal policy and denote it by π*.

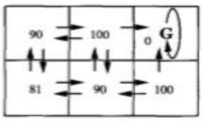
$$\pi^* \equiv \operatorname*{argmax}_{\pi} V^{\pi}(s), (\forall s)$$



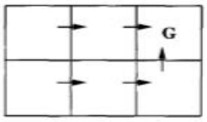
$r(s,a)$ (immediate reward) values



$Q(s,a)$ values



$V^*(s)$ values



One optimal policy

$$0+\gamma 100+\gamma^2 0+\gamma^3 0+\cdots=90$$

Q-learning

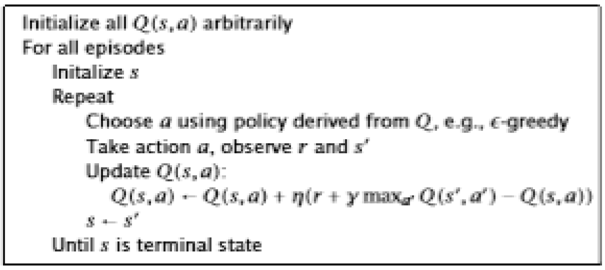
- we may have an imperfect robot which sometimes fails to go in the intended direction and deviates, or advances shorter or longer than expected.

In such a case, we have

$$Q(s_t,a_t)-E[r_{t+1}]+\gamma\sum_{s_{t+1}}P(s_{t+1}|s_t,a_t)\max_{a_{t+1}}Q(s_{t+1},a_{t+1})$$

- We cannot do a direct assignment in this case because for the same state and action, we may receive different rewards or move to different next states. What we do is keep a running average. This is known as the Q learning algorithm:

$$\hat{Q}(s_t,a_t)\leftarrow\hat{Q}(s_t,a_t)+\eta(r_{t+1}+\gamma\max_{a_{t+1}}\hat{Q}(s_{t+1},a_{t+1})-Q(s_t,a_t))$$



```
Initialize all $Q(s,a)$ arbitrarily
For all episodes
    Initalize $s$
    Repeat
        Choose $a$ using policy derived from $Q$, e.g., $\epsilon$-greedy
        Take action $a$, observe $r$ and $s'$
        Update $Q(s,a)$:
            $Q(s,a)\leftarrow Q(s,a)+\eta(r+\gamma\max_{a'}Q(s',a')-Q(s,a))$
        $s\leftarrow s'$
    Until $s$ is terminal state
```

Video Content / Details of website for further learning (if any):

https://youtu.be/DhdUlDIAG7Y

https://blog.floydhub.com/an-introduction-to-q-learning-reinforcement-learning/

Important Books/Journals for further learning including the page nos.:

EthemAlpaydin, “Introduction to Machine Learning”, Second Edition, MITPress,2013,Page no: 454-461

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L -10

CSE

IV/VII

Course Name with Code:Machine Learning Techniques -16CSE14

Course Teacher : Dr.N.Naveenkumar

Unit : V Date of Lecture:

Topic of Lecture: Fundamentals of Sentiment Analysis

Introduction: (Maximum 5 sentences):**Sentiment analysis** is the interpretation and classification of emotions (positive, negative and neutral) within text data using text **analysis** techniques

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)
Machine learning techniques.
Machine learning Algorithm implementation

Fundamentals of Sentiment Analysis

- **Sentiment analysis** allows businesses to identify customer **sentiment** toward products, brands or services in online conversations and feedback.

Sentimental Analysis Basics

- Sentiment analysis models detect polarity within a text (e.g. a *positive* or *negative* opinion), whether it's a whole document, paragraph, sentence, or clause.
- Understanding people's emotions is essential for businesses since customers are able to express their thoughts and feelings more openly than ever before.
- By automatically analyzing customer feedback, from survey responses to social media conversations, brands are able to listen attentively to their customers, and tailor products and services to meet their needs.

Types Of Sentiment Analysis

- Fine-grained Sentiment Analysis
- Emotion detection
- Aspect-based Sentiment Analysis
- Multilingual sentiment analysis

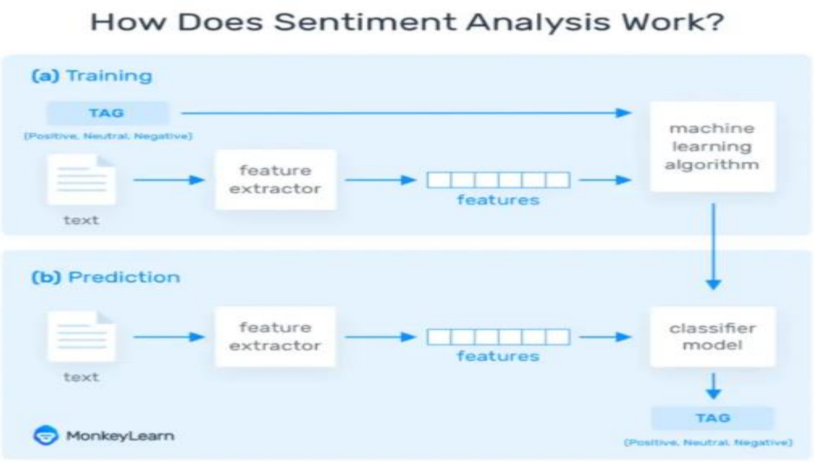
The Main Types Of Algorithms Used Include:

- **Rule-based** systems that perform sentiment analysis based on a set of manually crafted rules.

- **Automatic** systems that rely on machine learning techniques to learn from data.
- **Hybrid** systems that combine both rule-based and automatic approaches.

Example

- one of our customers used sentiment analysis to automatically analyze 4,000+ reviews about their product, and discovered that customers were happy about their pricing but complained a lot about their customer service


How Does Sentiment Analysis Work?
(a) Training
TAG
(Positive, Neutral, Negative)
text
feature extractor
features
machine learning algorithm
(b) Prediction
text
feature extractor
features
classifier model
TAG
(Positive, Neutral, Negative)
MonkeyLearn

Sentiment Analysis Challenges

- Subjectivity and Tone
- Context and Polarity
- Irony and Sarcasm
- Comparisons
- Defining Neutral

Sentiment Analysis Use Cases & Applications

- Social media monitoring
- Brand monitoring
- Voice of customer (VoC)
- Customer service
- Market research

Video Content / Details of website for further learning (if any):
https://youtu.be/AJVP96tAWxw

https://monkeylearn.com/sentiment-analysis/

Important Books/Journals for further learning including the page nos:
MATERIAL

Course Teacher

Verified by HOD