Machine Learning - Lecture Naïve Bayes Classifier

Prof. Dr. Dewan Md. Farid

Professor of Computer Science
United International University

June 13, 2023

Classifying Medical Image

- Let us first simulate a simplified case *mimicking* a medical image classification task.
- Figure 1 shows two images, each having a distinct region inside it. The region are also themselves visually different. We could say that the region of Figure 1(a) results from a benign lesion, class A, and that of Figure 1(b) from a malignant one (cancer), class B.
- We will further assume that these are not the only patterns (images) that are available to us., but we have access to an image database with a number of patterns.

Example of Medical Image

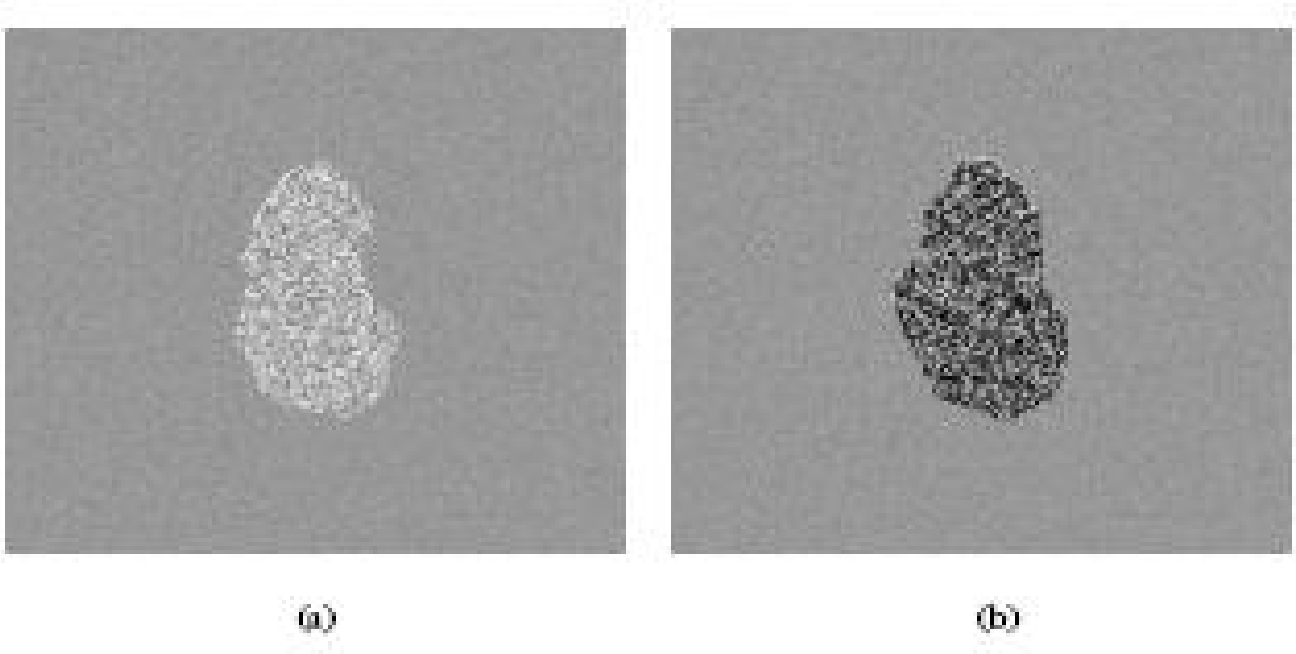

Figure: Regions corresponding to (a) class A and (b) class B.

Classifying Medical Image (con.)

- The first step is to identify the measurable quantities that makes these two regions *distinct* from each other.
- Figure 2 shows a plot of the mean value of the intensity in each region of interest versus the corresponding standard deviation around this mean.
- Each point corresponds to a different image from the available database.
- The straight line in Figure 2 is known as the *decision line* that seems to be a good candidate for separating the two classes.
- Let us now assume that we are given a new image with a region in it and that we do not know to which class it belongs, which is shown by the asterisk (*) in Figure 2.

Ploting Medical Images

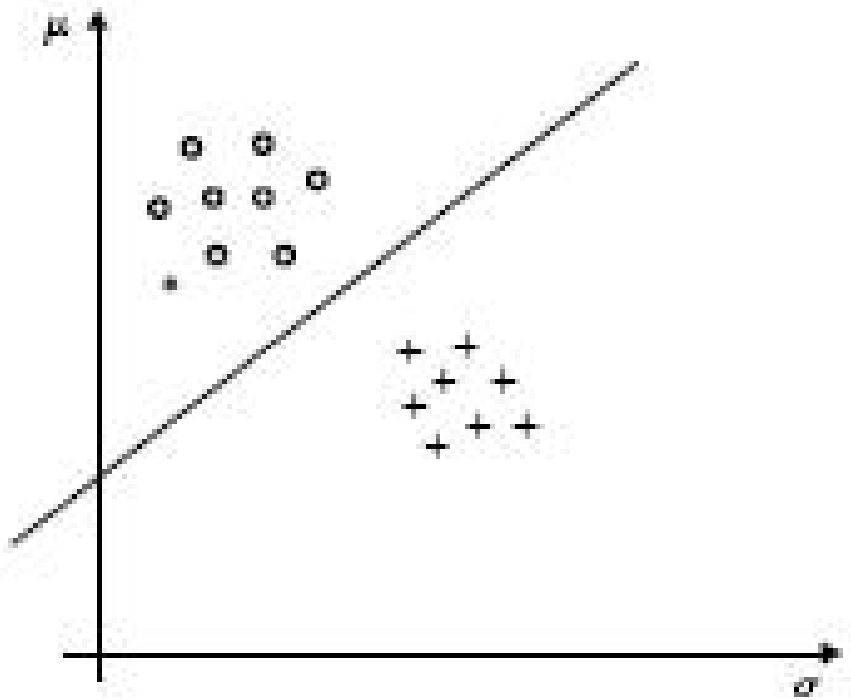

Figure: Plot of the mean value versus the standard deviation for a number of different images originating from class A(o) and class B(+).

Decision Line

The measurements used for the classification, the mean value and the standard deviation in this case, are known as *features*. In the more general case l features x_i, $i = 1, 2, \cdots, l$ are used, and they form the *feature vector*.

$$x = [x_1, x_2, \cdots, x_l]^T \tag{1}$$

Where T denotes transposition. Each of the feature vectors identifies *uniquely* a single pattern (object). The **decision line** in Figure 2 constitutes the *classifier* whose role is to divide the feature space into regions that correspond to either class A or class B. The basic stages involved in the design of a classification system is shown in Figure 2.

The Design of a Classification System

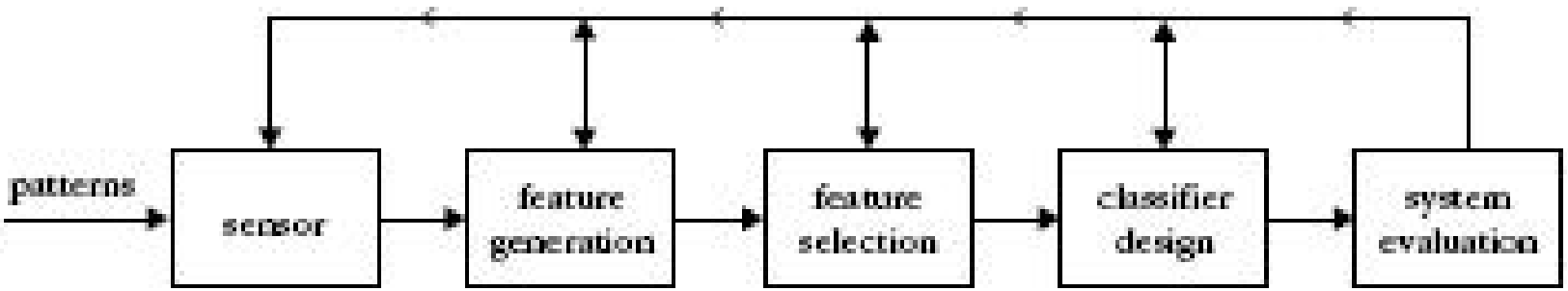

Figure: The basic stages involved in the design of a classification system.

Bayes Decision Theory

- We will design a classifier using Bayes decision theory that classify an unknown pattern in the most probable of the classes.
- Given a classification task of M classes, $\omega_1, \omega_2, \cdots, \omega_M$, and an unknown pattern which is represented by a feature vector x, we form the M conditional probabilities $P(\omega_i|x), i = 1, 2, \cdots, M$. Sometimes, these are also referred to as a *posteriori probabilities*.
- In words, each of them represents the probability that the unknown pattern belongs to the respective class ω_i, given that the corresponding feature vector takes the value x. The unknown pattern is then assigned to the class corresponding to the **maximum** of these M.
- The Bayes decision theory estimates the **Probability Density Functions** (pdf), based on the available experimental evidence, that is, the feature vectors corresponding to the patterns of the training set.

Priori Probabilities

- We will initially focus on the two-class case. Let ω_1 and ω_2 be the two classes in which our patterns belong.
- First we need to estimate the **priori probabilities**, $P(\omega_1)$ and $P(\omega_2)$ from the available training feature vectors.
- If N is the total number of available training patterns, and N_1, N_2 of them belong to ω_1 and ω_2, respectively, the $P(\omega_1) \approx \frac{N_1}{N}$ and $P(\omega_2) \approx \frac{N_2}{N}$.

Class-Conditional Probabilities

- Second we need to known are the class-conditional probability density functions, $P(x|\omega_i, i = 1, 2)$, describing the distribution of the feature vectors in each of the classes.
- The pdf $P(x|\omega_i)$ is sometimes referred to as the *likelihood function* of ω_i with respect to x.
- Now we have all the ingredients to compute the conditional probabilities using the *Bayes rule*:

$$P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)} \tag{2}$$

Where $P(x)$ is the pdf of x.

$$P(x) = \sum_{i=1}^{2} P(x|\omega_i)P(\omega_i) \tag{3}$$

Bayes classification rule

The *Bayes classification rule* can now stated as: if Equ. 4, then x is classified to ω_1, or if Equ. 5, then x is classified to ω_2.

$$P(\omega_1|x) > P(\omega_2|x) \tag{4}$$

$$P(\omega_1|x) < P(\omega_2|x) \tag{5}$$

The case of equality is detrimental and the pattern can be assigned to either of the two classes if it is like Equ. 6.

$$P(x|\omega_1)P(\omega_1) \gtrless P(x|\omega_2)P(\omega_2) \tag{6}$$

Bayes classification rule (con.)

$P(x)$ is not taken into account, because it is the same for all classes and it does not affect the decision. Furthermore, if the a *priori probabilities* are equal, that is Equ. 7.

$$P(\omega_1) = P(\omega_2) = \frac{1}{2} \tag{7}$$

$$P(x|\omega_1) \gtrless P(x|\omega_2) \tag{8}$$

Equiprobable Classes

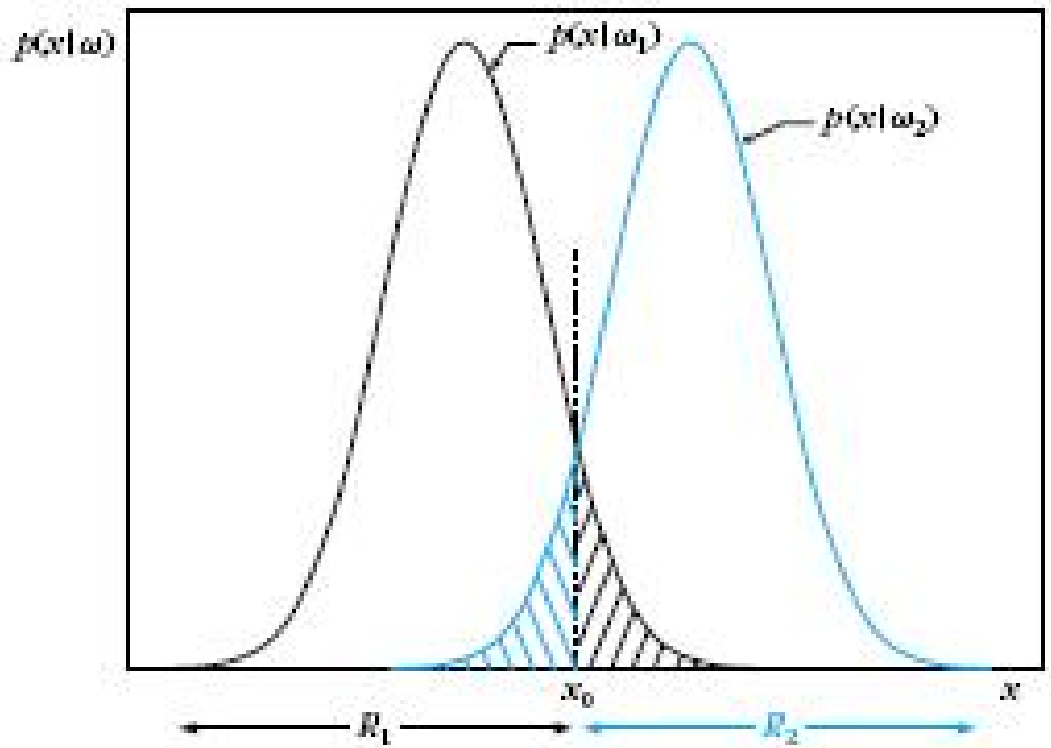

Figure: Example of the two regions R_1 and R_2 formed by the Bayesian classifier for the case of two equiprobable classes.

Equiprobable Classes (con.)

- Figure 4 presents an example of two equiprobable classes and shows the variations of $P(x|\omega_i)$, $i = 1, 2$ as functions of x for the simple case of a single feature($l = 1$).
- The dotted line at x_0 is a threshold partitioning the feature space into two regions R_1 and R_2.
- According to the Bayes decision rule, for all values of x in R_1 the classifier decides ω_1 and for all values in R_2 it decides ω_2.
- However, it is obvious from the figure that decision errors are unavoidable. Indeed, there is a finite probability for an x to lie in the R_2 region and at the same time to belong in class ω_1. Then our decision is in error. The same is true for points originating from class ω_2.

Naïve Bayes Classifier

- A naïve Bayes (NB) classifier is a simple probabilistic classifier based on: (a) Bayes theorem, (b) strong (naïve) independence assumptions, and (c) independent feature models.
- It is also an important mining classifier for pattern classification and applied in many real world classification problems because of its high classification performance.
- A NB classifier can easily handle missing attribute values by simply omitting the corresponding probabilities for those attributes when calculating the likelihood of membership for each class.
- The NB classifier also requires the class conditional independence, i.e. the effect of an attribute on a given class is independent of those of other attributes.

Advantages of NB classifier

The naïve Bayes (NB classifier has several advantages such as:

1. Easy to use.
2. Only one scan of the training data required.
3. Handling missing attribute values.
4. Continuous data.
5. High classification performance.

Dataset

- Given a training dataset, $D = \{X_1, X_2, \cdots, X_n\}$, each data record is represented as, $X_i = \{x_1, x_2, \cdots, x_n\}$.
- D contains the following attributes $\{A_1, A_2, \cdots, A_n\}$ and each attribute A_i contains the following attribute values $\{A_{i1}, A_{i2}, \cdots, A_{ih}\}$.
- The attribute values can be discrete or continuous.
- D also contains a set of classes $C = \{C_1, C_2, \cdots, C_m\}$. Each training instance, $X \in D$, has a particular class label C_i.
- For a test instance, X, the classifier will predict that X belongs to the class with the highest posterior probability, conditioned on X.

NB classifier

The NB classifier predicts that the instance X belongs to the class C_i, if and only if $P(C_i|X) > P(C_j|X)$ for $1 \leq j \leq m, j \neq i$. The class C_i for which $P(C_i|X)$ is maximized is called the Maximum Posteriori Hypothesis.

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{9}$$

In Bayes theorem shown in Eq. 9, as $P(X)$ is a constant for all classes, only $P(X|C_i)P(C_i)$ needs to be maximized. If the class prior probabilities are not known, then it is commonly assumed that the classes are equally likely, that is, $P(C_1) = P(C_2) = \cdots = P(C_m)$, and therefore maximize $P(X|C_i)$. Otherwise, maximize $P(X|C_i)P(C_i)$. The class prior probabilities are calculated by $P(C_i) = |C_{i,D}|/|D|$, where $|C_{i,D}|$ is the number of training instances belonging to the class C_i in D.

NB classifier (con.)

To compute $P(X|C_i)$ in a dataset with many attributes is extremely computationally expensive. Thus, the naïve assumption of class conditional independence is made in order to reduce computation in evaluating $P(X|C_i)$. The attributes are conditionally independent of one another, given the class label of the instance. Thus, Eq. 10 and Eq. 11 are used to produce $P(X|C_i)$.

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \tag{10}$$

$$P(X|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times \cdots \times P(x_n|C_i) \tag{11}$$

In Eq. 10, x_k refers to the value of attribute A_k for instance X. Therefore, these probabilities $P(x_1|C_i), P(x_2|C_i), \cdots, P(x_n|C_i)$ can be easily estimated from the training instances.

NB classifier (con.)

Moreover, the attributes in training datasets can be categorical or continuous-valued. If the attribute value, A_k, is categorical, then $P(x_k|C_i)$ is the number of instances in the class $C_i \in D$ with the value x_k for A_k, divided by $|C_{i,D}|$, i.e. the number of instances belonging to the class $C_i \in D$. If A_k is a continuous-valued attribute, then A_k is typically assumed to have a Gaussian distribution with a mean μ and standard deviation σ, defined respectively by the following two equations:

$$P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) \tag{12}$$

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-p)^2}{2\sigma^2}} \tag{13}$$

In Eq. 12, μ_{C_i} is the mean and σ_{C_i} is the standard deviation of the values of the attribute A_k for all training instances in the class C_i. Now we can bring these two quantities to Eq. 13, together with x_k, in order to estimate $P(x_k|C_i)$.

NB classifier (con.)

To predict the class label of instance X, $P(X|C_i)P(C_i)$ is evaluated for each class $C_i \in D$. The NB classifier predicts that the class label of instance X is the class C_i, if and only if

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \tag{14}$$

In Eq. 14, $1 \leq j \leq m$ and $j \neq i$. That is the predicted class label is the class C_i for which $P(X|C_i)P(C_i)$ is the maximum probability. Algorithm 1 outlines the naïve Bayes classifier algorithm.

Algorithm 1 Naïve Bayes classifier

Input: $D = \{x_1, x_2, \cdots, x_n\}$ // Training data.
Output: A naïve Bayes Model.
Method:

1: **for** each class, $C_i \in D$, **do**
2: Find the prior probabilities, $P(C_i)$.
3: **end for**
4: **for** each attribute, $A_i \in D$, **do**
5: **for** each attribute value, $A_{ij} \in A_i$, **do**
6: Find the class conditional probabilities, $P(A_{ij}|C_i)$.
7: **end for**
8: **end for**
9: **for** each instance, $x_i \in D$, **do**
10: Find the posterior probability, $P(C_i|x_i)$;
11: **end for**

Laplace Correction

- A zero probability cancels the effects of all the other (posteriori) probabilities (on C_i) involved in the product.
- We can assume that our training database, D, is so large that adding one to each count that we need would only make a negligible difference in the estimated probability value, yet would conveniently avoid the case of probability values of zero.
- This technique estimation is known as the **Laplacian correction** or **Laplace estimator**, named after Pierre Laplace, a French mathematician who lived from 1749 to 1827.

NB classifier - An Illustrative Example

To illustrate the operation of naïve Bayes, NB classifier, we consider a small dataset in Table 1 described by four attributes namely Outlook, Temperature, Humidity, and Wind, which represent the weather condition of a particular day. Each attribute has several unique attribute values. The Play column in Table 1 represents the class category of each instance. It indicates whether a particular weather condition is suitable or not for playing tennis.

Table: The playing tennis dataset

Day	Outlook	Temperature	Humidity	Wind	Play
D_1	Sunny	Hot	High	Weak	No
D_2	Sunny	Hot	High	Strong	No
D_3	Overcast	Hot	High	Weak	Yes
D_4	Rain	Mild	High	Weak	Yes
D_5	Rain	Cool	Normal	Weak	Yes
D_6	Rain	Cool	Normal	Strong	No
D_7	Overcast	Cool	Normal	Strong	Yes
D_8	Sunny	Mild	High	Weak	No
D_9	Sunny	Cool	Normal	Weak	Yes
D_{10}	Rain	Mild	Normal	Weak	Yes
D_{11}	Sunny	Mild	Normal	Strong	Yes
D_{12}	Overcast	Mild	High	Strong	Yes
D_{13}	Overcast	Hot	Normal	Weak	Yes
D_{14}	Rain	Mild	High	Strong	No

Table: Prior probabilities for each class generated using the playing tennis dataset

Probability	Value
$P(Play = Yes)$	$9/14 = 0.642$
$P(Play = No)$	$5/14 = 0.375$

Table: Conditional probabilities for Outlook calculated using the playing tennis dataset

Probability	Value
$P(Outlook = Sunny \vert Play = Yes)$	$2/9 = 0.222$
$P(Outlook = Sunny \vert Play = No)$	$3/5 = 0.6$
$P(Outlook = Overcast \vert Play = Yes)$	$4/9 = 0.444$
$P(Outlook = Overcast \vert Play = No)$	$0/5 = 0.0$
$P(Outlook = Rain \vert Play = Yes)$	$3/9 = 0.3$
$P(Outlook = Rain \vert Play = No)$	$2/5 = 0.4$

Table: Conditional probabilities for Temperature, Humidity, and Wind calculated using the playing tennis dataset

Probability	Value
$P(Temperature = Hot \mid Play = Yes)$	$2/9 = 0.222$
$P(Temperature = Hot \mid Play = No)$	$2/5 = 0.4$
$P(Temperature = Mild \mid Play = Yes)$	$4/9 = 0.444$
$P(Temperature = Mild \mid Play = No)$	$2/5 = 0.4$
$P(Temperature = Cool \mid Play = Yes)$	$3/9 = 0.333$
$P(Temperature = Cool \mid Play = No)$	$1/5 = 0.2$
$P(Humidity = High \mid Play = Yes)$	$3/9 = 0.333$
$P(Humidity = High \mid Play = No)$	$4/5 = 0.8$
$P(Humidity = Normal \mid Play = Yes)$	$6/9 = 0.666$
$P(Humidity = Normal \mid Play = No)$	$1/5 = 0.2$
$P(Wind = Weak \mid Play = Yes)$	$6/9 = 0.666$
$P(Wind = Weak \mid Play = No)$	$2/5 = 0.4$
$P(Wind = Strong \mid Play = Yes)$	$3/9 = 0.333$
$P(Wind = Strong \mid Play = No)$	$3/5 = 0.6$

Using these probabilities, we obtain
$P(D_1|Play = Yes) = P(Outlook = Sunny|Play = Yes) * P(Temperature = Hot|Play = Yes) * P(Humidity = High|Play = Yes) * P(Wind = Weak|Play = Yes) = 0.222 * 0.222 * 0.333 * 0.666 = 0.0109$.
Similarly
$P(D_1|Play = No) = P(Outlook = Sunny|Play = No) * P(Temperature = Hot|Play = No) * P(Humidity = High|Play = No) * P(Wind = Weak|Play = No) = 0.6 * 0.4 * 0.8 * 0.4 = 0.0768$.
To find the class, C_i, that maximises $P(X|C_i)P(C_i)$, we compute:
$P(D_1|Play = Yes)P(Play = Yes) = 0.0109 * 0.642 = 0.00699$
$P(D_1|Play = No)P(Play = No) = 0.0768 * 0.375 = 0.0288$
Therefore, the naïve Bayes classifier predicts $Play = No$ for instance D_1.
Note: As $P(Outlook = Overcast|Play = No) = \frac{0}{5} = 0$. So, all instances with $Outlook = Overcast$ will be Yes (D_3, D_7, D_{12}, and D_{13} will be Yes).

Bayesian Network

A **Bayesian network** is a *directed acyclic graph* (DAG), where the nodes correspond to random variables (features). Each node is associated with a set of conditional probabilities, $P(x_i|A_i)$, where x_i is the variable associated with the specific node and A_i is the set of its parents in the graph.

$$p(x) = p(x_1) \prod_{i=2}^{I} p(x_i|A_i) \tag{15}$$

where, $A_i \subseteq \{x_{i-1}, x_{i-2}, \cdots, x_I\}$

Bayesian Network (con.)

For example, let $I = 6$ and

$$p(x_6|x_5, \cdots, x_1) = p(x_6|x_5, x_4) \tag{16}$$

$$p(x_5|x_4, \cdots, x_1) = p(x_5|x_4) \tag{17}$$

$$p(x_4|x_3, x_2, x_1) = p(x_4|x_2, x_1) \tag{18}$$

$$p(x_3|x_2, x_1) = p(x_3|x_2) \tag{19}$$

$$p(x_2|x_1) = p(x_2) \tag{20}$$

Then, $A_6 = \{x_5, x_4\}$, $A_5 = \{x_4\}$, $A_4 = \{x_2, x_1\}$, $A_3 = \{x_2\}$, $A_2 = \emptyset$

Bayesian Network Graphical Model

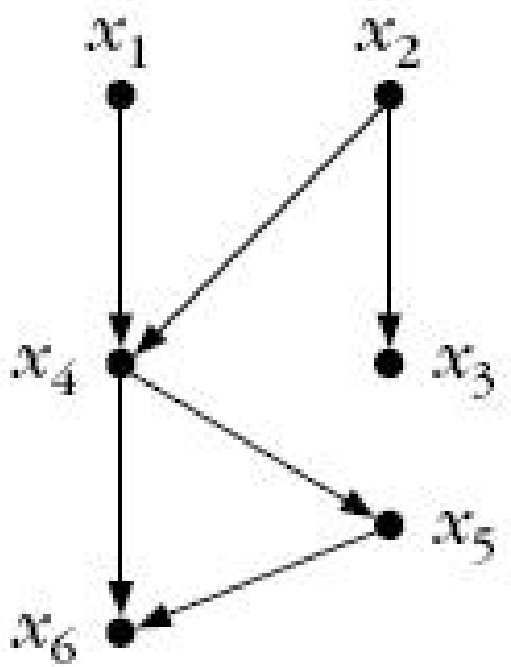

Figure: Graphical model - illustrating conditional dependencies.

Bayesina Network Modelling

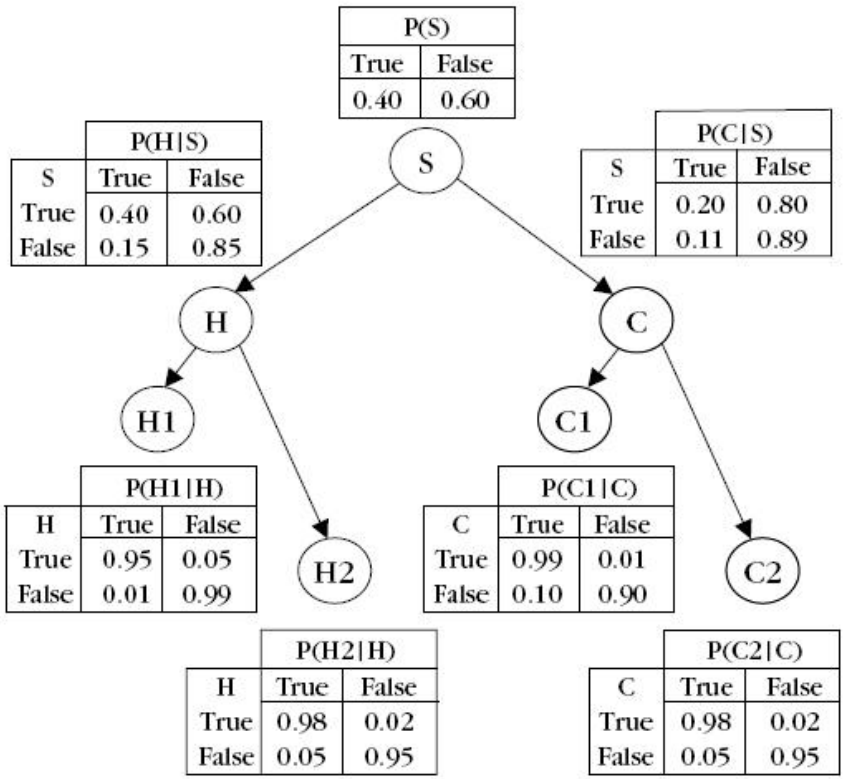

Figure: Conditional dependencies for an example concerning smokers (S), tendencies to develop cancer (c), and heart disease (H), together with variables corresponding to heart (H1, H2) and cancer (C1, C2) medical tests.

*** THANK YOU ***