Machine Learning

Bayesian Learning Methods

Introduction to Bayes Learning

What is Bayesian Learning?

Bayesian Learning is a probabilistic approach to inference.

Basic Assumptions:

- Quantities of interest are governed by probability distributions.
- Optimal decisions can be made by reasoning about these probabilities together with observed training data.

Motivation for Bayesian Learning

- The explicit manipulation of probabilities is among the most practical approaches to certain types of learning problems (e.g. decision tree and neural network learning).
- It provides a useful perspective for understanding learning methods that do not explicitly manipulate probabilities. For example:
  - Determining the conditions under which algorithms output the most probable hypothesis
  - The justification of the error functions in ANNs
  - The justification of the bias in decision trees

Meet the Reverend Bayes

- **Two main works:**
  - *Divine Benevolence, or an Attempt to Prove That the Principal End of the Divine Providence and Government is the Happiness of His Creatures* (1731)
  - *An Introduction to the Doctrine of Fluxions, and a Defence of the Mathematicians Against the Objections of the Author of the Analyst* (1736, anonymous)
- **We are interested in:**
  - ***Essay Towards Solving a Problem in the Doctrine of Chances* (1764)** which was published posthumously by Richard Price

Foundations of Bayesian Theory

Immanuel Kant (1724-1804)

- *Copernican revolution*: our understanding of the external world has its foundations in both experience and a priori concepts

Isaac Newton (1643-1727)

- *Universal gravitation*
- three *laws of motion* which dominated the scientific view of the physical universe for the next three centuries

Features of Bayesian Learning

- Each observed training instance is an example that can incrementally decrease or increase the estimated probability that a hypothesis is correct.
- Prior knowledge can be combined with observed data to determine the final probability of a hypothesis.

Features of Bayesian Learning (2)

- Hypotheses make probabilistic predictions.
- New instances can be classified by combining the predictions of multiple hypotheses, weighted by their probabilities.
- It provides a standard of optimal decision making against which other approaches can be measured.

Practical Difficulties

- Initial knowledge of many probabilities is required.
- There is significant computation costs required.

Bayes' Theorem

What is Bayes' Theorem?

- In machine learning, we are interested in finding the best hypothesis h from some space H, given the observed training data D.
- The best hypothesis is likely the most probable hypothesis.
- **Bayes' Theorem** provides a direct method of calculating the probability of such a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself.

Bayes' Theorem

Bayes' Theorem states:

$$P(h|D) = \frac{P(D|h)\,P(h)}{P(D)}$$

- $P(h)$ is the **prior probability of *h***, which reflects any background knowledge about the chance that h is correct.
- $P(D)$ us the **prior probability of *D***, which is the probability that D will be observed.
- $P(D|h)$ is the **probability of *D* given a world in which *h* holds**.
- $P(h|D)$ is the **posterior probability of *h***, which reflects the confidence that h holds after D has been observed.

Bayes' Theorem (2)

In plain English, using Bayesian probability terminology, Bayes' Theorem can be stated as:

$$posterior = \frac{prior \ * likelihood}{evidence}$$

The numerator is equivalent to the **joint probability model.**

Bayes' Theorem Example

A patient takes a lab test and the result comes back positive. It is known that the test returns a correct positive result in 98% of the cases and a correct negative result in 97% of the cases. Furthermore, only 0.008 of the entire population has this disease.

1. What is the probability that this patient has cancer?
2. What is the probability that she does not have cancer?
3. What is the your diagnosis?

Bayes' Theorem Example (2)

$P(+|\text{cancer}) = 0.98$
$P(\text{cancer}) = 0.008$
$P(+) = P(+|\text{cancer})\ P(\text{cancer}) + P(+|\neg\text{cancer})\ P(\neg\text{cancer})$
$= (0.98 * 0.008) + (0.03 * 0.992) = 0.0376$

$$P(\text{cancer}|+) = \frac{P(+|\text{cancer})\ P(\text{cancer})}{P(+)}$$

$$P(\text{cancer}|+) = \frac{0.98 \ * 0.008}{0.0376}$$

$$P(\text{cancer}|+) = \mathbf{0.2085}$$

Bayes' Theorem Example (3)

$P(+|\neg cancer) = 0.03$
$P(\neg cancer) = 0.992$
$P(+) = P(+|cancer)\ P(cancer) + P(+|\neg cancer)\ P(\neg cancer)$
$= (0.98 * 0.008) + (0.03 * 0.992) = 0.0376$

$$P(\neg cancer|+) = \frac{P(+|\neg cancer)\ P(\neg cancer)}{P(+)}$$

$$P(\neg cancer|+) = \frac{0.03\ * 0.992}{0.0376}$$

$$P(\neg cancer|+) = \mathbf{0.7915}$$

Bayes' Theorem (4)

Diagnosis:

$P(cancer|+) = \mathbf{0.2085}$
$P(\neg cancer|+) = \mathbf{0.7915}$

The patient does not have cancer!

MAP Hypothesis

- In many learning scenarios, the learner considers some set of candidate hypotheses H and is interested in finding the most probable hypothesis $h \in H$ given the observed training data D.
- This most probable hypothesis is called the **maximum a posteriori (MAP)** hypotheses.

MAP Hypothesis (2)

The MAP hypothesis is calculated as:

$$h_{MAP} = \underset{h \in H}{argmax}\, P(h|D)$$
$$= \underset{h \in H}{argmax}\, \frac{P(D|h)\, P(h)}{P(D)}$$
$$= \underset{h \in H}{argmax}\, P(D|h)\, P(h)$$

Note that $P(D)$ can be dropped in the denominator as it is a constant independent of h.

MAP Hypothesis Example

From the earlier cancer test example:

$P(+|\text{cancer}) = 0.98$	$P(-|\text{cancer}) = 0.02$
$P(+|\neg\text{cancer}) = 0.03$	$P(-|\neg\text{cancer}) = 0.97$
$P(\text{cancer}) = 0.008$	$P(\neg\text{cancer}) = 0.992$

$P(\text{cancer}|+) = P(+|\text{cancer})\, P(\text{cancer})$
$= 0.98 * 0.008 = \mathbf{0.0078}$

$P(\neg\text{cancer}|+) = P(+|\neg\text{cancer})\, P(\neg\text{cancer})$
$= 0.03 * 0.992 = \mathbf{0.0298}$

$h_{MAP} = \neg\text{cancer}$

ML Hypothesis

Sometimes it can be assumed that every hypothesis is equally probable a priori. In this case, the MAP hypothesis equation can be simplified further.

Because $P(D|h)$ is the likelihood of D given h, this is called the *maximum likelihood* (*ML*) hypothesis.

$$h_{ML} = \underset{h \in H}{argmax}\, P(D|h)$$

Note that $P(h)$ can be dropped as it is the same for each $h \in H$.

Most Probable Classification

- In Bayesian learning, the primary question is: What is the most probable hypothesis given data?
- We can also ask: For a new test point, what is the most probable classification given the training data?
- *Is classification the same as the prediction of the maximum a posteriori hypothesis?*

Most Probable Classification (2)

Suppose our hypothesis space H has three functions h_1, h_2 and h_3:

- $P(h_1|D) = 0.4$
- $P(h_2|D) = 0.3$
- $P(h_3|D) = 0.3$

What is the MAP hypothesis? h_1

Most Probable Classification (3)

For a new instance $\boldsymbol{x}$, suppose:

$h_1(\boldsymbol{x}) = +1$ $h_2(\boldsymbol{x}) = -1$ $h_3(\boldsymbol{x}) = -1$

What is the most probable classification?

$P(+1|\boldsymbol{x}) = 0.4$ $P(-1|\boldsymbol{x}) = 0.3 + 0.3 = 0.6$

The most probable classification is not the same as the prediction of the MAP hypothesis!

Bayes Optimal Classifier

The **Bayes Optimal Classification** is defined as the category produced by the most probable classifier:

$$\underset{y}{argmax} \sum_{h_i \in H} P(y|h_i)\, P(h_i|D)$$

- Computing this can be hopeless inefficient
- It is an interesting theoretical concept nonetheless since no other classification method can beat it on average

Bayes Optimal Classifier (2)

What should *H* be?

H can be a collection of functions

- Given the training data, choose an optimal function
- Then, given new data, evaluate the selected function on it

H can be a collection of possible predictions

- Given the data, try to directly choose the optimal prediction

Naïve Bayes Classifier

What is a Naïve Bayes Classifier?

- A **Naive Bayes classifier** is one of a collection of classification algorithms based on Bayes' Theorem.
- It is not a single algorithm but a family of algorithms where all of them share a common principle: *the naïve Bayes assumption*.

Advantages of Naïve Bayes

- Despite their basic design and oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations.
- Naive Bayes classifiers only require a small number of training data to estimate the parameters necessary for classification.

Naïve Assumption

The fundamental **Naive Bayes assumption** is that each feature's contribution to the predicted outcome is:

- **independent**
- **equal**

The assumptions made by Naive Bayes classifiers are not generally correct in real world situations. The independence assumption is *never* correct but often works well in practice.

Tennis Data

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Understanding "Naïve"

Consider the Tennis data. Naïve means that:

- No pair of features are dependent. For example, the temperature being 'Hot' has nothing to do with the outlook being 'Rainy'. Hence, the features are assumed to be **independent**.
- Each feature is given the same weight (or importance). For example, knowing only temperature and humidity alone can't predict the outcome accurately. None of the attributes are irrelevant and all are assumed to be contributing **equally** to the outcome.

Bayes' Theorem (redux)

We can now apply Bayes' Theorem in the following way:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

where y is class variable and X is a dependent feature vector $X = (x_1,x_2,x_3,...,x_n)$.

For example, day 1 of the tennis data would give Y = no and X = (Sunny, Hot, High, Weak).

Bayes' Theorem (redux) (2)

If any two events are independent then

$$P(\text{A},\text{B}) = P(\text{A})\ P(\text{B})$$

Hence,

$$P(y|x_1x_2\ldots,x_n) = \frac{P(x_1|y)\ P(x_2|y)\ldots P(x_n|y)\ P(y)}{P(x_1)\ P(x_2)\ldots P(x_n)}$$

Bayes Theorem (redux) (3)

The previous equation can be re-expressed as:

$$P(y|x_1x_2\ldots,x_n) = \frac{P(y)\prod_i^n P(x_i|y)}{P(x_1)\ P(x_2)\ldots P(x_n)}$$

As the denominator remains constant for any given input, that term may be removed as follows:

$$P(y|x_1x_2\ldots,x_n) \propto P(y)\prod_i^n P(x_i|y)$$

where $\propto$ denotes proportionality.

Naïve Bayes Classifier

The next step is the creation of a **classifier model.**

For this, we find the probability of a given set of inputs for all possible values of the class variable *y* and choose the output with maximum probability.

This can be expressed mathematically as:

$$y = \underset{y}{argmax}\ P(y)\prod_i^n P(x_i|y)$$

Naïve Bayes Classifier (2)

Next we calculate $P(y)$ and $P(x_i|y)$.

$P(y)$ is sometimes called the **class probability**.
$P(x_i|y)$ is called the **conditional probability**.

The various naïve Bayes classifiers differ mainly by the assumptions they make regarding the distribution $P(x_i|y)$.

Example

Let us apply the above formula manually to our tennis dataset. For this, we need to do a set of precomputations.

We need to find $P(x_i \mid y_j)$ for each x_i in X and each y_j in Y.

Example (2)

Lots of calculations later…

P(yes) = 9/14 = 0.643	P(no) = 5/15 = 0.333
P(wind=strong | yes) = 3/9 = 0.333	P(wind=strong | no) = 3/5 = 0.6
P(wind=weak | yes) = 6/9 = 0.667	P(wind=weak | no) = 3/5 = 0.4
P(humid=high | yes) = 3/9 = 0.333	P(humid=high | no) = 4/5 = 0.8
P(humid=normal | yes) = 6/9 = 0.667	P(humid=normal | no) = 1/5 = 0.2
P(temp=hot | yes) = 2/9 = 0.222	P(temp=hot | no) = 2/5 = 0.4
P(temp=mild | yes) = 4/9 = 0.444	P(temp=mild | no) = 2/5 = 0.4
P(temp=cool | yes) = 3/9 = 0.333	P(temp=cool | no) = 1/5 = 0.2
P(outlook=sunny | yes) = 2/9 = 0.222	P(outlook=sunny | no) = 3/5 = 0.6
P(outlook=overcast | yes) = 4/9 = 0.444	P(outlook=overcast | no) = 0/5 = 0
P(outlook=rain | yes) = 3/9 = 0.333	P(outlook=rain | no) = 2/5 = 0.4

Example (3)

Classify the following datum:

x=(outlook=sunny, temp=cool, humid=high, wind=strong)

Will your roommate play tennis or not?

Yes?

P(yes) P(outlook=sunny | yes) P(temp=cool | yes) P(humid=high | yes) P(wind=strong | yes)

0.643*0.222*0.333*0.333*0.333 = **0.00527**

No?

P(no) P(outlook=sunny | no) P(temp=cool | no) P(humid=high | no) P(wind=strong | no)

0.333*0.6*0.2*0.8*0.6 = **0.0192**

Answer: NO

"…probability theory is more fundamentally concerned with the *structure* of reasoning and causation than with numbers."

- Glenn Shafer and Judea Pearl

Introduction to Readings in Uncertain Reasoning

Bayesian Belief Networks

Why Bayesian Networks?

- Naïve Bayes classifiers make significant use of the assumption of conditional independence.
- Conditional independence dramatically reduces the complexity of a learning task.
- However, in many cases this assumption is overly restrictive.

Bayesian Networks

A **Bayesian Network (BN)** or a **Bayesian Belief Network** is a type of probabilistic graphical model that uses Bayesian inference for probability computations.

The Bayesian network represents a set of variables and their conditional probabilities with a *directed acyclic graph* (*DAG*).

Notation

- Bayesian Networks describe the probability distribution over a set of variables $x_1, x_2, \ldots, x_n$.
- The **joint space** of the network is:
  $$V(x_1) \times V(x_2) \times \ldots \times V(x_n)$$
- The **joint probability distribution** is the probability for each of the possible variable bindings for the tuple $(x_1, x_2, \ldots, x_n)$.

Representation

Nodes:

- Each node represents a variable in the joint space of the problem.
- For each node a conditional probability table is given that describes the probability distribution of the variable given the values of its immediate predecessors.

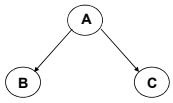
Edges:

- A directed edge between nodes represents a conditional dependence between the originator and the destination.
- No edge between nodes represents conditional independence.

Marginal Independence

$P(A,B,C) = P(A)\ P(B)\ P(C)$

Conditional Independence

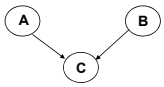


$P(A,B,C) = P(B|A)\ P(C|A)\ P(A)$

B and C are conditionally independent given A.

e.g., A is a disease, and B and C as conditionally independent symptoms given A.

Conditional Dependence



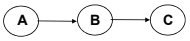
Independent Causes:
$P(A,B,C) = P(C|A,B)\ P(A)\ P(B)$

A and B are marginally independent, but become dependent once C is known.

Given C, observing A makes B less likely.

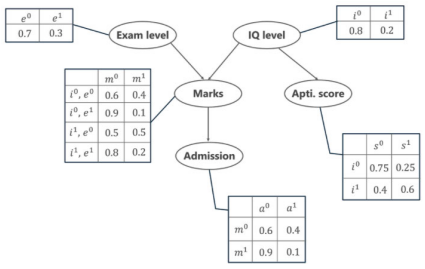
e.g., A=earthquake, B=burglary, C=alarm

Markov Dependence



$P(A,B,C) = P(C|B)\ P(B|A)\ P(A)$

A Bayesian Network



Joint Probability Distribution

By definition, the probabilities of all different possible combinations of a set of variables is called its **Joint Probability Distribution (JPD)**.

Consider four variables x_1, x_2, x_3 and x_4.

$P(x_1,x_2,x_3,x_4)$ is the JPD of these variables and is calculated as:

$P(x_1,x_2,x_3,x_4) = P(x_1|x_2,x_3,x_4)\ P(x_2,x_3,x_4)$

$P(x_1,x_2,x_3,x_4) = P(x_1|x_2,x_3,x_4)\ P(x_2|x_3,x_4)\ P(x_3,x_4)$

$P(x_1,x_2,x_3,x_4) = P(x_1|x_2,x_3,x_4)\ P(x_2|x_3,x_4)\ P(x_3|x_4)\ P(x_4)$

This is called the **Chain Rule for Probabilities**.

Joint Probability Distribution (2)

For *n* variables, this process can be generalized into the formula:

$$P(x_1, x_2, \ldots, x_n) = \sum_{i}^{n} x_i \mid Parents(x_i)$$

where $Parents(x_i)$ denotes the set of immediate predecessors of x_i.

JPD Examples

Case 1: Calculate the probability that in spite of the exam level being difficult and the student having a low IQ level and a low aptitude score, he manages to pass the exam and secure admission to the university.

From the above word problem statement, the Joint Probability Distribution is:

P(a=1, m=1, i=0, e=1, s=0)

JPD Examples (2)

From the conditional probability tables shown in the Bayesian belief graph, the joint probability distribution is calculated as follows:

P(a=1, m=1, i=0, e=0, s=0)
= *P*(a=1 | m=1) *P*(m=1 | i=0, e=1) *P*(i=0) *P*(e=1) *P*(s=0 | i=0)
= 0.1 * 0.1 * 0.8 * 0.3 * 0.75
= **0.0018**

JPD Examples (3)

Case 2: In another case, calculate the probability that if the student has a high IQ level and aptitude score and the exam is easy, she still fails to pass and does not secure admission to the university.

The formula for the JPD is given by:

$P(a=0, m=0, i=1, e=0, s=1)$

JPD Examples (4)

From the conditional probability tables shown in the Bayesian belief graph, the joint probability distribution is calculated as follows:

$P(a=0, m=0, i=1, e=0, s=1)$
$= P(a=0 \mid m=0)\, P(m=0 \mid i=1, e=0)\, P(i=1)\, P(e=0)\, P(s=1 \mid i=1)$
$= 0.6 * 0.5 * 0.2 * 0.7 * 0.6$
$= \mathbf{0.0252}$

Bayesian Network Inference

- If the values are known for all other variables in the network, inference is straightforward.
- In the more general case, values are only known for a subset of the network variables.
- A Bayesian Network can be used to compute the probability distribution for any subset of network variables given the values or distributions for any subset of the remaining variables.

Bayesian Network Inference (2)

network structure: known or unknown?

network variables: observable or partially observable?

- For a known structure and fully observable variables, the conditional probabilities can be estimated as for a naive Bayes classifier.
- For a known structure and partially observable variables, the learning problem can be compared to learning weights for an ANN.
- For an unknown structure, heuristic algorithms or scoring metric can be used.

Constructing a BN

There are two ways to build a Bayesian Belief Network:

1. Manual construction
2. Automatic construction

Both methods have advantages and disadvantages.

Manual Construction of BNs

Manual construction of a Bayesian network assumes prior expert knowledge of the underlying domain.

There are two fundamental steps in the construction:

1. build the DAG
2. Assess the conditional probability distribution in each node

Manual Construction of BNs (2)

Building the DAG:

1. Identify the random variables that are the nodes in the BN.
   - Not all variables have to be observed.
   - Some random variables may actually specify unobserved quantities that are believed to influence the observable outcomes.

Manual Construction of BNs (3)

Building the DAG:

2 Identify the structural dependencies between variables.
   - The graph structure is usually based on subject or expert knowledge.
   - Model criticism and revision are often essential.

Manual Construction of BNs (4)

Assess the conditional probability distribution in each node:

- If the variables are discrete, this can be represented as a table which lists the probability that the child node takes on each of its different values for each combination of values of its parents.
- If the conditional probability distribution is not available, other statistical methods such as *frequency estimation* can be used.

Estimating Probabilities

Normally, probabilities are estimated by the fraction of times the event is observed to occur n_c divided by the total number of opportunities to observe n:

$$\frac{n_c}{n}$$

In most cases this method is a good estimate, but if n_c is very small it provides poor results:

- It becomes a biased underestimate of the probability.
- If this estimate equals zero, it will dominate the Bayesian classifier.

Estimating Probabilities (3)

In the absence of information, it is common to assume a uniform distribution for p

$$p = \frac{1}{k}$$

where k is the number of attribute values

Estimating Probabilities (2)

The Bayesian approach to estimation is called the ***m-estimate***:

$$\frac{n_c + mp}{n + m}$$

where p is a prior estimate of the probability we wish to determine, and m is a constant called the equivalent sample size which determines how heavily to weight p relative to the observed data.

Automatic Construction of BNs

Bayesian networks may be learnt automatically straight from databases using experience-based algorithms often built in to appropriate software.

- Most automatic learning algorithms require that no data be missing in the dataset.
- There has to be enough data to satisfy the algorithm's requirements for reliable estimates of the conditional probability distributions.
- This is an NP-hard problem!