BAYESIAN MODEL

3rd Sem, MCA

Contents

- Concept Learning
- Bayesian Leaning
- Naïve Bayesian Classifier

Concept learning

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	YES
2	Sunny	Warm	High	Strong	Warm	Same	YES
3	Rainy	Cold	High	Strong	Warm	Change	NO
4	Sunny	Warm	High	Strong	Warm	Change	YES

ATTRIBUTES CONCEPT

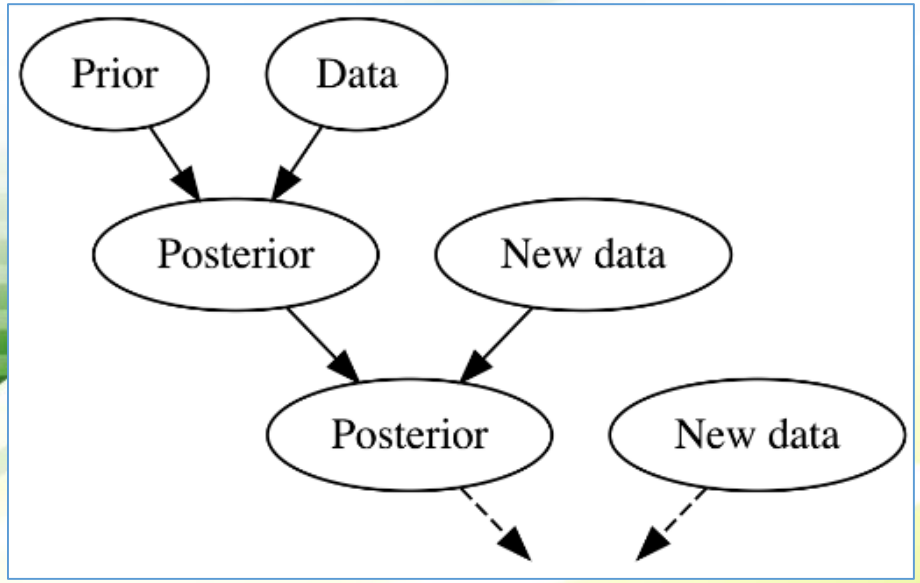
- Inducing general functions from specific training examples is a main issue of machine learning.
- Concept learning describes the process by which experience allows us to partition objects in the world into classes for the purpose of generalization, discrimination, and inference.
- Concept Learning is Acquiring the definition of a general category from given sample positive and negative training examples of the category.
- Concept Learning can be seen as a problem of searching through a predefined space of potential hypotheses for the hypothesis that best fits the training examples.
- The hypothesis space has a general-to-specific ordering of hypotheses, and the search can be efficiently organized by taking advantage of a naturally occurring structure over the hypothesis space.

Bayesian learning

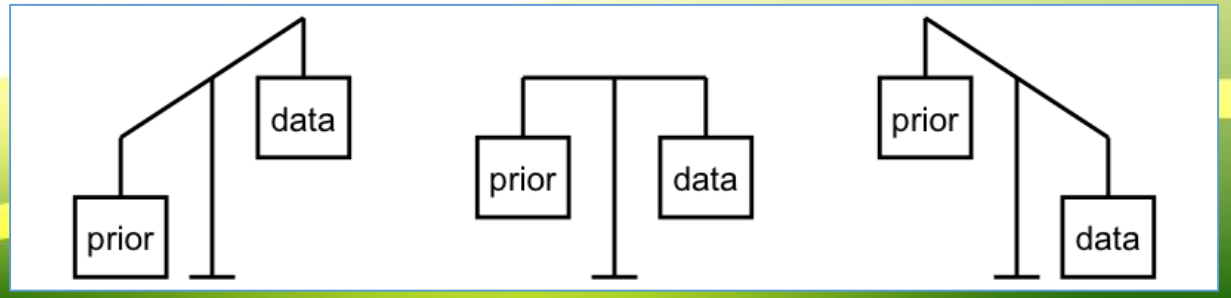
- In Bayesian concept learning, Prior knowledge can be combined with observed data to determine the final probability of a hypothesis.
- In machine learning, we try to determine the best hypothesis from some hypothesis space H, given the observed training data D.
- In Bayesian learning, the best hypothesis means the most probable hypothesis, given the data D plus any initial knowledge about the prior probabilities of the various hypotheses in H.
- Bayes theorem provides a way to calculate the probability of a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself.

Bayesian learning

- Bayesian knowledge-building diagram.



- Bayesian philosophy provides a formal framework that depends upon prior information, data, and the balance between them.



Bayesian learning

P(h) is prior probability of hypothesis h

- P(h) denote the initial probability that hypothesis h holds, before observing training data.
- P(h) may reflect any background knowledge about the chance that h is correct.

P(D) is prior probability of training data D

- The probability of D given no knowledge about which hypothesis holds

P(h|D) is posterior probability of h given D

- reflects influence of training data D, in contrast to the prior probability P(h), which is independent of D.

P(D|h) is posterior probability of D given h

- The probability of observing data D given some world in which hypothesis h holds.

Bayesian learning

- Bayes theorem provides a way to calculate the posterior probability P(h|D), from the prior probability P(h), together with P(D) and P(D|h).

$$\textbf{Bayes Theorem:} \quad P(h \mid D) = \frac{P(D \mid h)\, P(h)}{P(D)}$$

- P(h|D) increases with P(h) and P(D|h) according to Bayes theorem.
- P(h|D) decreases as P(D) increases, because the more probable it is that D will be observed independent of h, the less evidence D provides in support of h.

Posterior probability — Likelihood — Prior probability

$$p(h \mid d) = \frac{p(d \mid h)\, p(h)}{\sum_{h' \in H} p(d \mid h')\, p(h')}$$

Bayesian learning

Consider the given data.

A holds	T	T	F	F	T	F	T
B holds	T	F	T	F	T	F	F

$P(A) = 4/7 \qquad P(B) = 3/7 \qquad P(B|A) = 2/4 \qquad P(A|B) = 2/3$

$P(B|A) = P(A|B)P(B) / P(A) = (2/3 * 3/7) / 4/7 = 2/4$ → CORRECT

$P(A|B) = P(B|A)P(A) / P(B) = (2/4 * 4/7) / 3/7 = 2/3$ → CORRECT

Bayesian learning

Product rule: probability P(A ∧ B) of a conjunction of two events A and B:

$$P(A \wedge B) = P(A|B)P(B) = P(B|A)P(A)$$

Sum rule: probability of disjunction of two events A and B:

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

Theorem of total probability: if events A1,…,An are mutually exclusive with $\sum_{i=1}^{n} P(A_i) = 1$

$$P(B) = \sum_{i=1}^{n} P(B \mid A_i) P(A_i)$$

Bayesian learning: MAP

- The learner considers some set of candidate hypotheses H and it is interested in finding the most probable hypothesis h in H given the observed data D.
- Any such maximally probable hypothesis given the training data is called a **maximum a posteriori (MAP) hypothesis h_{MAP}**.
- We can determine the MAP hypotheses by using Bayes theorem to calculate the posterior probability of each candidate hypothesis.

$$h_{MAP} = \arg\max_{h \in H} P(h \mid D)$$
$$= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)}$$
$$= \arg\max_{h \in H} P(D \mid h) P(h)$$

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

Bayesian learning: ML

- If we assume that every hypothesis in H is equally probable

  i.e. **$P(h_i) = P(h_j)$ for all h_i and h_j in H**

  then, we can only consider P(D|h) to find the most probable hypothesis.

- P(D|h) is often called the **likelihood of the data D given h.**

- Any hypothesis that maximizes P(D|h) is called a **maximum likelihood (ML) hypothesis, h_{ML}.**

$$h_{ML} = \arg\max_{h_i \in H} P(D \mid h_i)$$

Bayesian learning: MAP

A patient takes a lab test and the result comes back positive. The test returns a correct positive result in only 98% of the cases in which the disease is actually present, and a correct negative result in only 97% of the cases in which the disease is not present. Furthermore, 0.8% of the entire population have this cancer.

P(cancer) = .008 **P(notcancer) = .992**
P(+|cancer) = .98 **P(-|cancer) = .02**
P(+|notcancer) = .03 **P(-|notcancer) = .97**

A patient takes a lab test and the result comes back positive.

P(+|cancer) P(cancer) = .98 * .008 = .0078
P(+|notcancer) P(notcancer) = .03 * .992 = .0298 → **h_{MAP} is notcancer**

Since P(cancer|+) + P(notcancer|+) must be 1
P(cancer|+) = .0078 / (.0078+.0298) = .21
P(notcancer|+) = .0298 / (.0078+.0298) = .79

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

$$h_{MAP} = \arg\max_{h \in H} P(h \mid D)$$
$$= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)}$$
$$= \arg\max_{h \in H} P(D \mid h) P(h)$$

Bayesian learning: BF MAP

- **Brute-Force Bayes Concept Learning Algorithm** finds the maximum posteriori hypothesis (h_{MAP}), based on Bayes theorem. ***(Brute-Force MAP Learning Algorithm / BF MAP algorithm)***

  1. For each hypothesis h in H, calculate the posterior probability. $$P(h \mid D) = \frac{P(D \mid h)\, P(h)}{P(D)}$$
  2. Find the Output the hypothesis h_{MAP} with the highest posterior probability. $$h_{MAP} = \arg\max_{h \in H} P(h \mid D)$$

- This algorithm may require *significant computation*, because it applies Bayes theorem to each hypothesis in H to calculate P(h|D) .
  - may be impractical for large hypothesis spaces,
  - algorithm is still of interest because it provides a standard against which we may judge the performance of other concept learning algorithms.

Bayesian learning: Consistent Learner

- A learning algorithm is a *consistent learner* if it outputs a hypothesis that commits zero errors over the training examples.
- Every consistent learner outputs a MAP hypothesis, if we assume
  - a uniform prior probability distribution over H

    i.e., $P(h_i) = P(h_j)$ for all i, j and
  - deterministic, noise free training data

    i.e., $P(D|h) = 1$ if D and h are consistent, and 0 otherwise.

Bayesian learning: Bayes Optimal Classifier

- The most probable classification of the new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities.
- If possible classification of new example can take on any value v_j from some set V, then probability $P(v_j \mid D)$ that the correct classification for the new instance is v_j :

$$P(v_j|D) = \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)$$

Bayes optimal classification:

$$\arg\max_{v_j \in V} \sum_{h_i \in H} P(v_j \mid h_i)P(h_i \mid D)$$

- Bayes optimal classifier obtains best performance at a cost; because the expense is due to the fact that it computes posterior probability for every hypothesis in H and then combines predictions of each hypothesis to classify each new instance.

Bayesian learning: Bayes Optimal Classifier

Bayes Optimal Classifier Example

$$P(h_1 \mid D) = .4, \quad P(- \mid h_1) = 0, \quad P(+ \mid h_1) = 1$$
$$P(h_2 \mid D) = .3, \quad P(- \mid h_2) = 1, \quad P(+ \mid h_2) = 0$$
$$P(h_3 \mid D) = .3, \quad P(- \mid h_3) = 1, \quad P(+ \mid h_3) = 0$$

Probabilities:

$$\sum_{h_i \in H} P(+ \mid h_i) P(h_i \mid D) = .4$$
$$\sum_{h_i \in H} P(- \mid h_i) P(h_i \mid D) = .6$$

Bayes Optimal Classifier Result

$$\arg\max_{v_j \in V} \sum_{h_i \in H} P(v_j \mid h_i) P(h_i \mid D) = \text{-}$$

Bayesian learning

- Bayes theorem can be derived from the conditional probability:

$$P(X/Y) = \frac{P(X \cap Y)}{P(Y)}, \text{ if } P(Y) \neq 0$$

$$P(Y/X) = \frac{P(Y \cap X)}{P(X)}, \text{ if } P(X) \neq 0$$

- P (X∩Y) is **joint probability** of both X and Y being true.
- P(Y) = P(Y | X) * P(X) + P(Y | not X) * P(not X)
- **complement of X:** P(not X) = 1 – P(X)

$$P(Y \cap X) = P(X \cap Y)$$

$$\text{or, } P(X \cap Y) = P(X/Y)P(Y) = P(Y/X)P(X)$$

$$\text{or, } P(X/Y) = \frac{P(Y/X)P(X)}{P(Y)}, \text{ if } P(Y) \neq 0$$

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

$$h_{MAP} = \arg\max_{h \in H} P(h \mid D)$$

Bayesian learning

Bag1 contains 4 white and 8 black balls and Bag2 contains 5 white and 3 black balls.

From one of the bag one ball is drawn at random and the ball which is drawn comes out as black.

Find the probability that the ball is drawn from Bag1.

Let E1, E2 and A be three events,
E1 = Event of selecting Bag1
E2 = Event of selecting Bag2
A = Event of drawing black ball

P(E1) = P(E2) = 1/2
P(drawing a black ball from Bag1) = P(A|E1) = 8/12 = 2/3
P(drawing a black ball from Bag2) = P(A|E2) = 3/8

P(A) = P(drawing a black ball) = P(A|E1) * P(E1) + P(A|E2) * P(E2)
= 2/3 * 1/2 + 3/8 * 1/2 = 25/48

According to Bayes' Theorem,
Probability(drawing a black ball from Bag1)
P(E1|A) = P(A|E1) * P(E1) / P(A)

According to Bayes' Theorem,
Probability(drawing a black ball from Bag1)
P(E1|A) = P(A|E1) * P(E1) / P(A)
= (2/3 * 1/2) / (25/48) = 16/25

Probability that ball is drawn from Bag1 is 16/25

Bayesian learning

According to a research of publicly-traded companies, 60% of the companies that increased their share price by more than 5% in the last three years replaced their CEOs during the period.

At the same time, only 35% of the companies that did not increase their share price by more than 5% in the same period replaced their CEOs.

Knowing that the probability that the stock prices grow by more than 5% is 4%, find the probability that the shares of a company that fires its CEO will increase by more than 5%.

Define the notation of probabilities.
P(I) –probability that stock price increases by 5% = (4%) = 0.04
P(R | I) –probability of CEO replacement given stock price has increased by 5% = (60%) = 0.6
P(I | R) –probability of stock price increases by 5% given that CEO has been replaced = ???

P(I`) = 1 – 0.4 = 0.6
P(R | I`) = (35%) = 0.35

According to Bayes' Theorem,
P(I | R) = P(R | I) * P(I) / P(R)

P(R) = P(R | I) * P(I) + P(R | I`) * P(I`)

$$P(A|B) = \frac{0.60 \times 0.04}{0.60 \times 0.04 + 0.35 \times (1 - 0.04)} = 0.067 \text{ or } 6.67\%$$

Bayesian learning

SpamAssassin works as a mail filter to identify spam in which users train the system. In emails, it considers patterns in words which are marked as spam by users.

For Example, it may have learned that the word “release” is marked as spam in 30% of the emails.

Concluding 0.8% of non-spam mails which includes the word “release” and 40% of all emails which are received by user is spam.

Find probability that a mail is a spam if the word “release” seems in it.

P(R | S) = 0.30
P(R | N) = 0.008
P(S) = 0.40
=> P(N) = 0.60

P(S | R) = ?

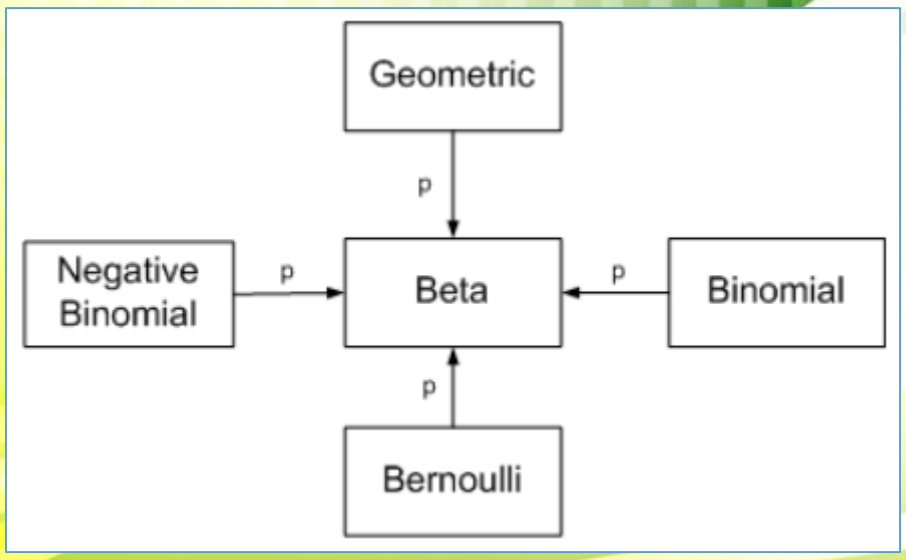
Now, using Bayes’ Theorem:
P(S | R) = P(R | S) * P(S) / P(R)

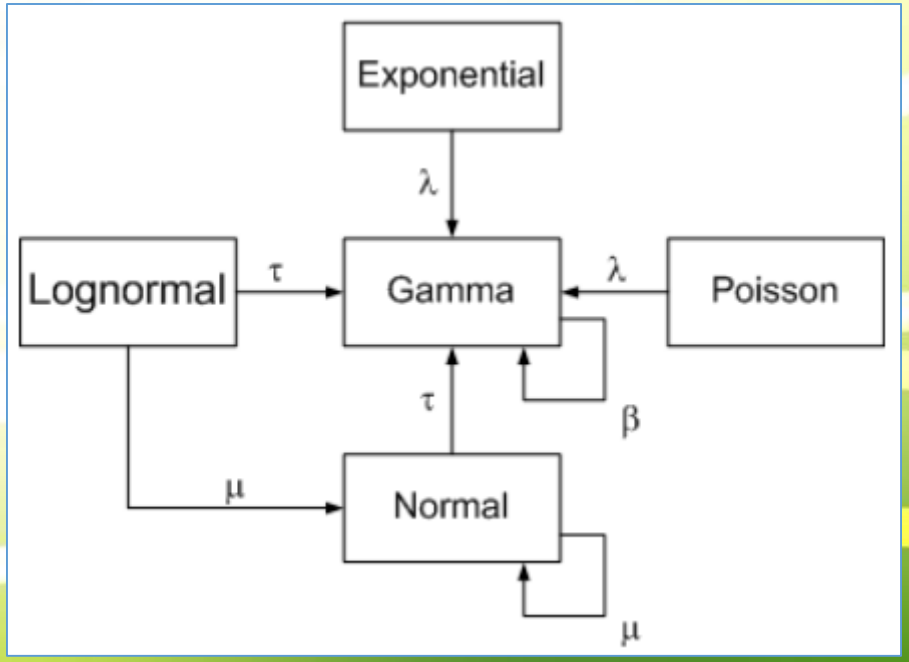
P(R) = P(R | S) * P(S) + P(R | N) * P(N)
= 0.40 * 0.30 + 0.30 *
0.008 = 0.1224

Using Bayes’ Theorem:
P(S | R) = P(R | S) * P(S) / P(R)
= 0.30 * 0.40 / 0.1224 = 0.980

Bayesian learning: Conjugate distribution

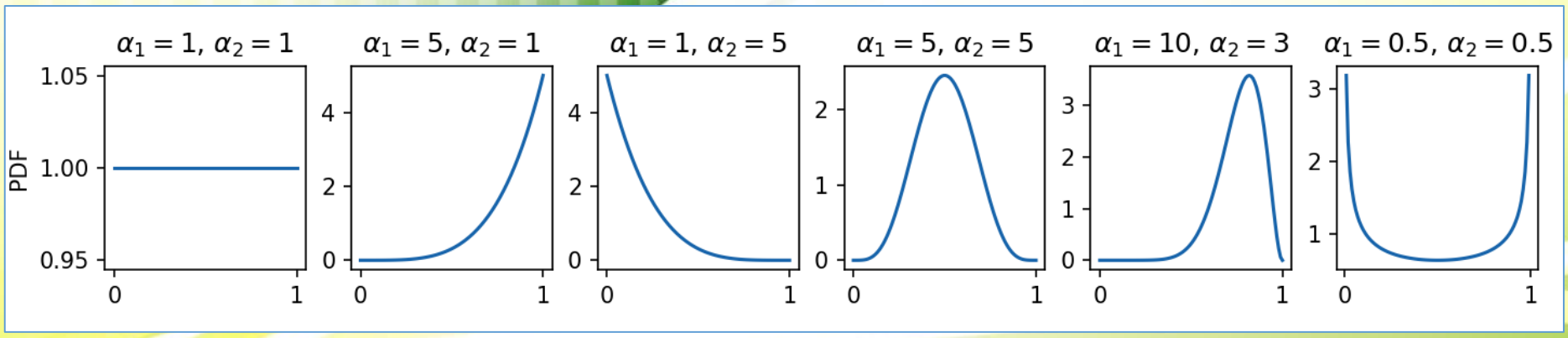
- In Bayesian probability theory, if the posterior distributions p(θ | x) are in the same probability distribution family as the prior probability distribution p(θ), the prior and posterior are then called ***conjugate distributions***, and the prior is called a conjugate prior for the likelihood function.





Bayesian learning: Beta distribution

- Beta distribution is a family of continuous probability distributions on the interval [0,1], often used as a prior for probabilities.
- Beta distribution has two parameters ***(α & β or α_1 & α_2)*** that control the shape of the distribution.



- Probability density function (PDF) of beta distribution

$$\text{beta}(x \mid \alpha_1, \alpha_2) = \frac{1}{\text{B}(\alpha_1, \alpha_2)} x^{\alpha_1 - 1}(1-x)^{\alpha_2 - 1}, \quad \alpha_1, \alpha_2 > 0.$$

$$\text{B}(\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_1)\ldots\Gamma(\alpha_K)}{\Gamma(\alpha_1 + \cdots + \alpha_K)}.$$

$$\Gamma(z) = (z-1)!$$

Bayesian learning: Beta-binomial model

- Beta-binomial model (along with closely-related beta-Bernoulli model) is one of the simplest Bayesian model.
- While *Binomial* distribution represents probability of success in a series of Bernoulli trials, *Beta* distribution here represents prior probability distribution of probability of success for each trial. *(conjugate distributions)*
- Thus, the probability p of a coin landing on head is modeled to be Beta distributed (with parameters α and β), while the likelihood of heads and tails is assumed to follow a Binomial distribution with parameters n (representing the number of flips) and the Beta-distributed p, thus creating the link.

$$p \sim Beta(\alpha,\beta)$$

$$y \sim Binomial(n,p)$$

Bayesian learning: Beta-binomial model

- Beta-binomial model assumes that each observation, y_i, represents number of successes from n_i Bernoulli trials for which probability of success, p_i, is drawn from a beta distribution.
- Mean and dispersion parameters for beta distribution, μ_i and φ_i, are then modelled as linear combinations of fixed and random effects.
- We then assume that each observation follows a binomial distribution, $Y_i \sim \text{Binomial}(n_i, p_i)$, where $pi \sim \text{Beta}(\mu_i, \varphi_i)$

$$Y|\pi \sim \text{Bin}(n, \pi)$$
$$\pi \sim \text{Beta}(\alpha, \beta)$$

Bayesian learning: Dirichlet-multinomial model

- Multinomial (i.e. multiple choices) distribution is a generalization of the Binomial distribution (i.e. binary choice).
- Dirichlet distribution is a generalization of the Beta distribution.
  - while Beta distribution models probability of a single probability p, Dirichlet models probabilities of multiple, mutually exclusive choices, parameterized by $\boldsymbol{\alpha}$ (concentration parameter that represents the weights for each choice.

$$\theta \sim Dirichlet(a)$$

$$y \sim Multinomial(n,\theta)$$

- *Example, coins for Beta-Binomial distribution and dice for Dirichlet-Multinomial distribution.*

Bayesian learning: Dirichlet-multinomial model

- Multivariate beta distribution is called Dirichlet distribution.
- General PDF of the Dirichlet distribution is

$$\mathrm{Dir}(\boldsymbol{\alpha}) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} x_1^{(\alpha_1-1)} \dots x_K^{(\alpha_K-1)}, \quad \alpha_1, \dots, \alpha_K > 0$$

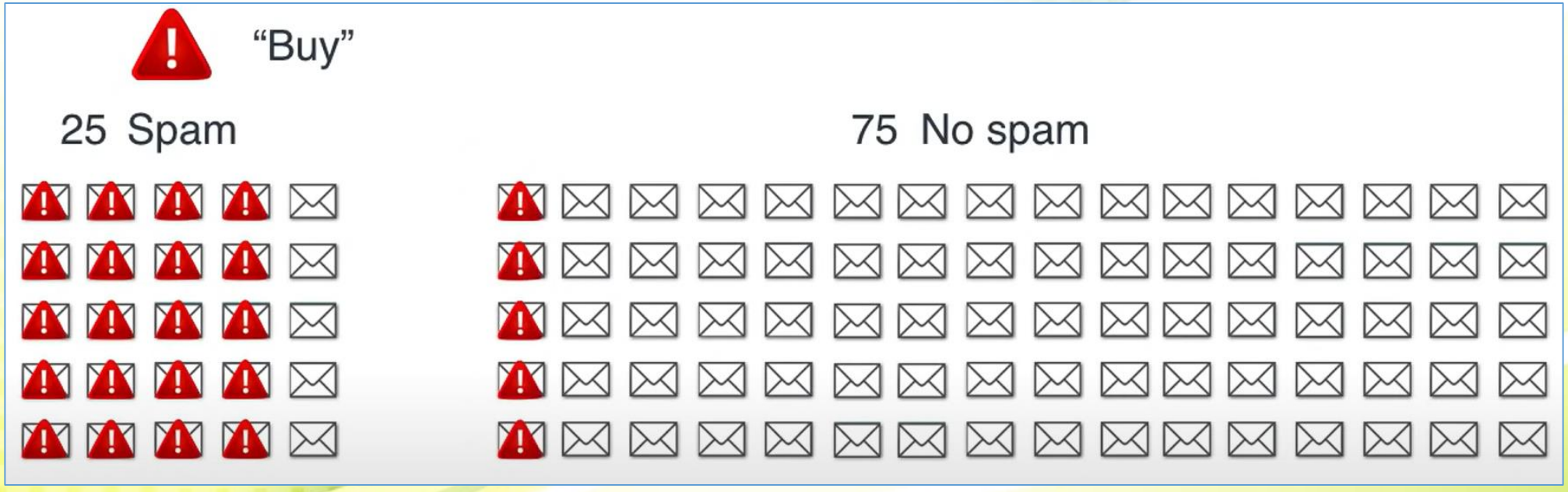
$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_1) \dots \Gamma(\alpha_K)}{\Gamma(\alpha_1 + \dots + \alpha_K)}.$$

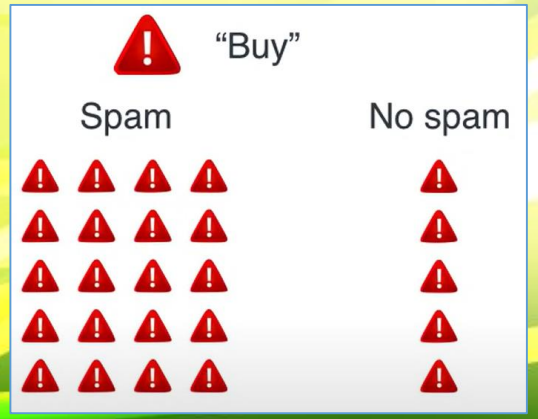
$$\Gamma(z) = (z-1)!$$

Naive Bayes Classifier

- Supervised learning algorithm, used for solving **classification** problems.
- Based on **Bayes' Theorem**.
- Naive Bayes classifier assumes that presence of a feature in a class is not related to any other feature.
  - Every pair of features being classified is *(conditionally)* independent of each other.
- Classification algorithm for both binary and multi-class classification problems.
  - *Work with only categorical response variables and Large data sets.*
- **Naive**: Called Naïve because it **assumes** that occurrence of a certain feature is (conditionally) independent of occurrence of other features. *(niave assumption)*
  - If fruit is identified based on color, shape, and taste; then red, spherical, and sweet fruit is recognized as apple.
  - Here each feature individually contributes to identify that it is an apple *without depending on each other.*
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.

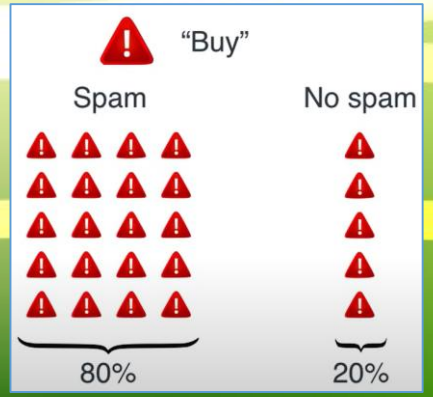
Naive Bayes Classifier



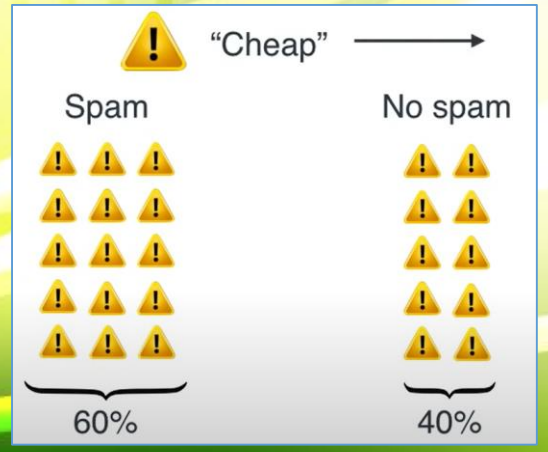


Quiz: If an e-mail contains the word "buy", what is the probability that it is spam?

- ○ 40%
- ○ 60%
- ○ 80%
- ○ 100%



Naive Bayes Classifier

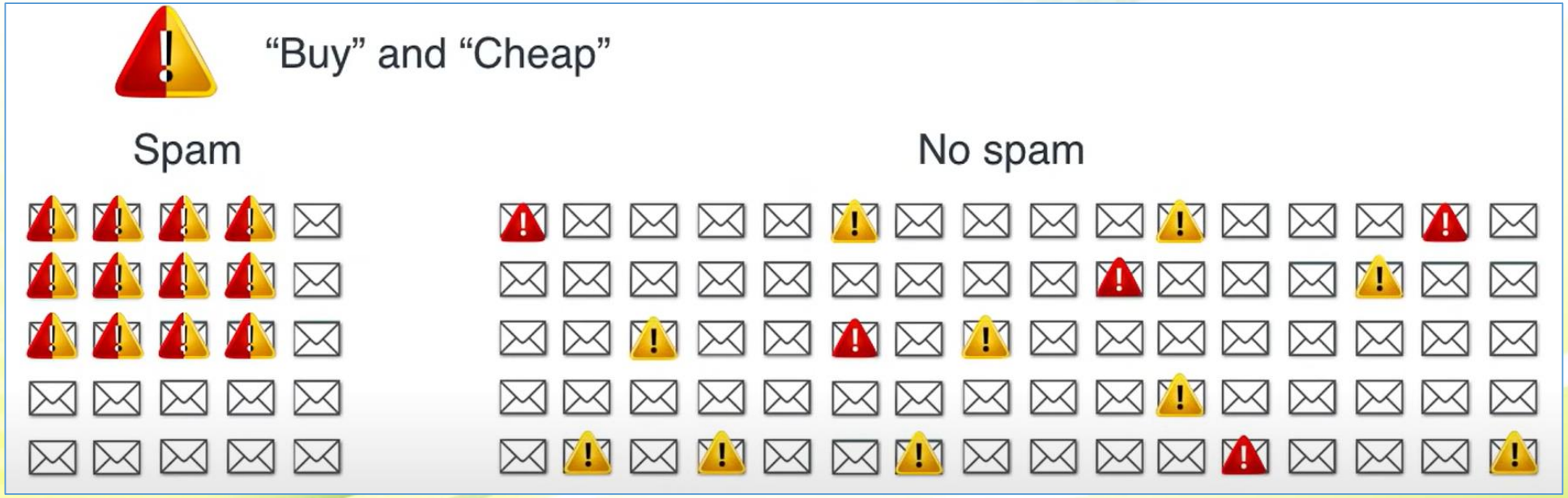


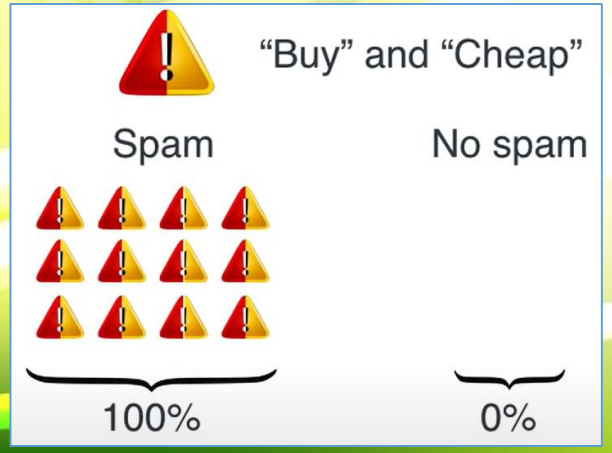
Quiz: If an e-mail contains the word "cheap", what is the probability that it is spam?

- ○ 40%
- ● 60%
- ○ 80%
- ○ 100%

Solution:
60%

Naive Bayes Classifier



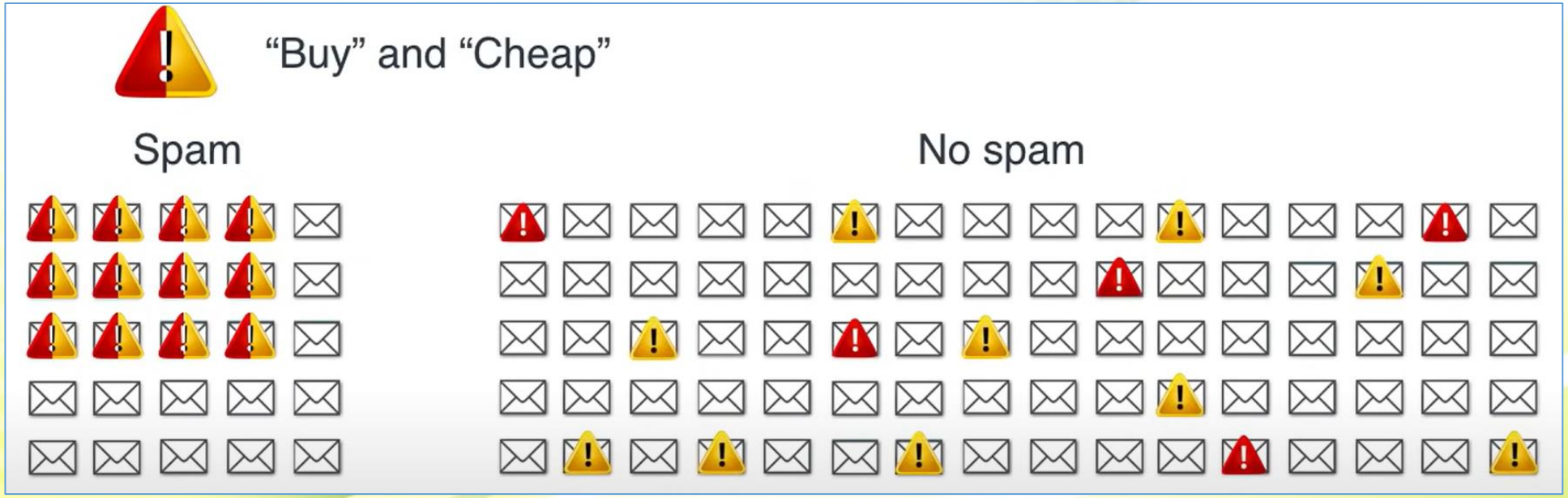


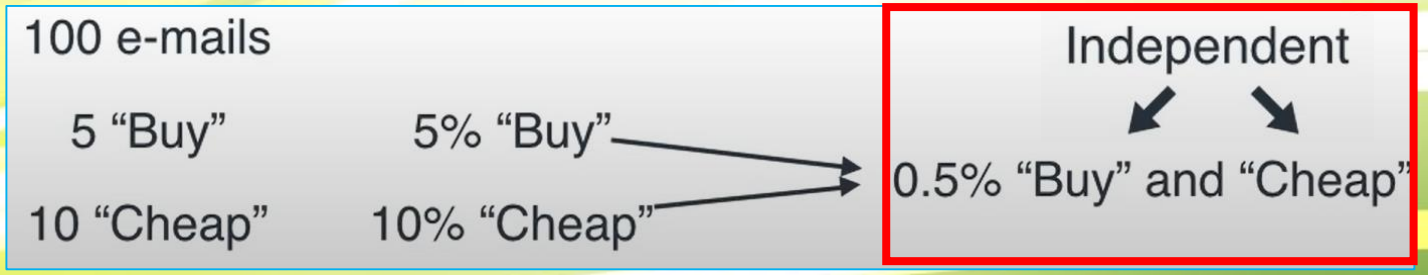
Quiz: If an e-mail contains the words "buy" and "cheap", what is the probability that it is spam?

- ○ 40%
- ○ 60%
- ○ 80%
- ● 100%

Solution:
100%

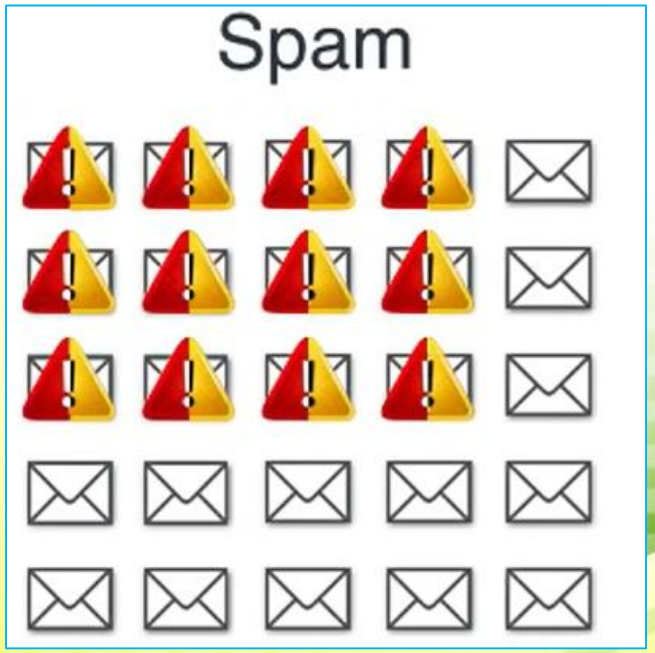
Naive Bayes Classifier

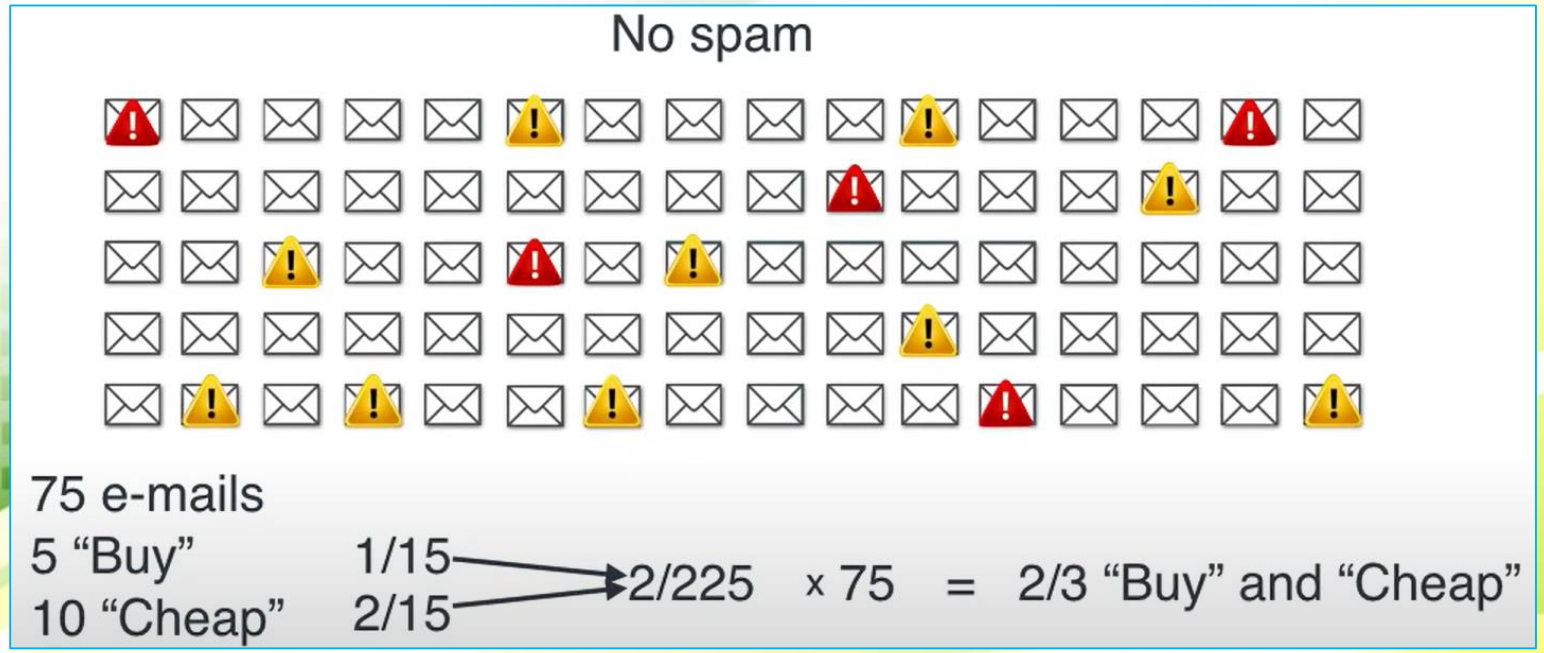




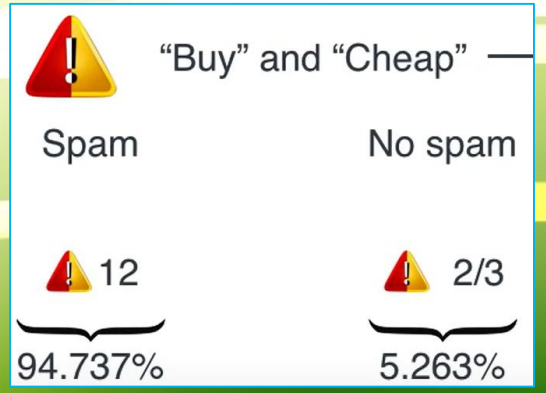
Naive

Naive Bayes Classifier





25 e-mails
20 “Buy”
15 Cheap
4/5
3/5
12/25 x 25 = 12 “Buy” and “Cheap”



Naive Bayes Classifier

$$P(S|B) = \frac{P(B|S)\,P(S)}{P(B|S)\,P(S) + P(B|H)\,P(H)}$$

P("Buy" & "Cheap") = P("Buy") P("Cheap")

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

↑ Naive

$$P(S|B \cap C) = \frac{P(B|S)\,P(C|S)\,P(S)}{P(B|S)\,P(C|S)\,P(S) + P(B|H)\,P(C|H)\,P(H)}$$

Naive Bayes Classifier

	Spam		No spam	
Total	25		75	
Buy	20	4/5	5	1/15
Cheap	15	3/5	10	2/15
Buy & Cheap	12	12/25	2/3	2/225

$$\frac{12}{12 + 2/3} = \frac{36}{38} = 94.737\%$$

Naive Bayes Classifier

	Spam		No Spam	
Total	25		75	
Buy	20	4/5	5	1/15
Cheap	15	3/5	10	2/15
Work	5	1/5	30	6/15
Buy, Cheap, & Work	12/5	12/125	4/15	12/3375

$$\frac{12/5}{12/5 + 4/15} = \frac{36}{40} = 90\%$$

Naive Bayes Classifier

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Joint Probability = P(A ∩ B) = P(A) x P(B)

Using naive conditional independence assumption for all i, this relationship is simplified to

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

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

$$P(y \mid x_1, \ldots, x_n) \propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow$$

$$\hat{y} = \arg\max_y P(y)\prod_{i=1}^{n} P(x_i \mid y),$$

Naive Bayes Classifier

- Supervised learning algorithm, used for solving **classification** problems.
- Naive Bayes classifier assumes that presence of a feature in a class is not related to any other feature (conditionally independent).
- Naive Bayes classifiers are 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.
  - **Gaussian Naïve Bayes** model assumes that features follow a normal distribution. This means if predictors take continuous values instead of discrete, then the model assumes that these values are sampled from the Gaussian distribution
  - **Multinomial Naïve Bayes** classifier is used when the data is multinomial distributed. It is primarily used for document classification problems, it means a particular document belongs to which category such as Sports, Politics, education, etc. The classifier uses the frequency of words for the predictors
  - **Bernoulli Naïve Bayes** classifier works similar to the Multinomial classifier, but the predictor variables are the independent Booleans variables. Such as if a particular word is present or not in a document. This model is also famous for document classification tasks.

Naive Bayes Classifier

Frequency Table

		Stolen?	
		Yes	No
Color	Red	3	2
	Yellow	2	3

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Color	Red	3/5	2/5
	Yellow	2/5	3/5

Frequency Table

		Stolen?	
		Yes	No
Type	Sports	4	2
	SUV	1	3

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Type	Sports	4/5	2/5
	SUV	1/5	3/5

Frequency Table

		Stolen?	
		Yes	No
Origin	Domestic	2	3
	Imported	3	2

Likelihood Table

		Stolen?	
		P(Yes)	P(No)
Origin	Domestic	2/5	3/5
	Imported	3/5	2/5

Color	Type	Origin	Stolen
Red	SUV	Domestic	?

P(Yes | X) = P(Red | Yes) * P(SUV | Yes) * P(Domestic | Yes) * P(Yes)

= 3/5 * 1/5 * 2/5 * 1

= 0.048

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

P(No | X) = P(Red | No) * P(SUV | No) * P(Domestic | No) * P(No)

= 2/5 * 3/5 * 3/5 * 1

= 0.144

Since 0.144 > 0.048 → for given features RED SUV & Domestic, it gets classified as 'NO' → car is not stolen.

Naive Bayes Classifier

- Consider the problem of playing golf.
- Build a predictive model whether the day is suitable for playing golf, given the features of the day

	Outlook	Temperature	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	High	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	Yes
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	No
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	Normal	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Naive Bayes Classifier

- Dataset is divided into two parts, **feature matrix** and **response vector**.
- Feature matrix contains all **dependent features** 'Outlook', 'Temperature', 'Humidity' and 'Windy'.
- Response vector contains value of **class variable** (output) 'Play golf' for each row of feature matrix.
- Naive Bayes goes with assumption that each feature makes an **independent** and **equal** contribution to outcome.

	Outlook	Temperature	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	High	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	Yes
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	No
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	Normal	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

- i.e. No pair of features are dependent.
  - Temperature being 'Hot' has nothing to do with humidity or outlook being 'Rainy' has no effect on winds.
- Secondly, each feature is given same weight (importance).
  - Knowing only temperature and humidity alone can't predict outcome **accurately**.
  - None of the attributes is irrelevant and assumed to be contributing equally to the outcome.
- *Assumptions made by Naive Bayes are not generally correct in real-world situations, but in theoretical concepts only*.

Naive Bayes Classifier

Bayes theorem can be rewritten as:

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

Variable **y** is output/response (play golf) → whether suitable to play or not given the conditions.

Variable **X** represent parameters/features.

$$X = (x_1, x_2, x_3, ..., x_n)$$

x_1,x_2….x_n represent features → outlook, temperature, humidity, windy.

By substituting for X and expanding using Chain rule;

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

	Outlook	Temperature	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	High	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	Yes
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	No
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	Normal	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Naive Bayes Classifier

- Consider the problem of playing golf.
- Build a predictive model whether the day is suitable for playing golf, given the weather is "Sunny".

P(Y|S) P(~Y|S)

P(Y|S) = P(S|Y)*P(Y)/P(S)
P(~Y|S) = P(S|~Y)*P(~Y)/P(S)

Frequency table for Weather Conditions:

Weather	Yes	No
Overcast	5	0
Rainy	2	2
Sunny	3	2
Total	10	5

Likelihood table weather condition:

Weather	No	Yes	
Overcast	0	5	5/14= 0.35
Rainy	2	2	4/14=0.29
Sunny	2	3	5/14=0.35
All	4/14=0.29	10/14=0.71	

P(Sunny|Yes)= 3/10= 0.3 P(Sunny)= 0.35 P(Yes)=0.71

Applying Bayes'theorem:

P(Yes|Sunny)= P(Sunny|Yes)*P(Yes)/P(Sunny) = 0.3*0.71/0.35= **0.60**

P(Sunny|NO)= 2/4=0.5 P(Sunny)= 0.35 P(No)= 0.29

P(No|Sunny)= P(Sunny|No)*P(No)/P(Sunny) = 0.5*0.29/0.35 = **0.41**

P(Yes|Sunny) > P(No|Sunny) → Hence on a Sunny day, Player can play game.

	Outlook	Play
0	Rainy	Yes
1	Sunny	Yes
2	Overcast	Yes
3	Overcast	Yes
4	Sunny	No
5	Rainy	Yes
6	Sunny	Yes
7	Overcast	Yes
8	Rainy	No
9	Sunny	No
10	Sunny	Yes
11	Rainy	No
12	Overcast	Yes
13	Overcast	Yes

Naive Bayes Classifier

Consider the problem of playing golf. Build a predictive model whether the day is suitable for playing (y), given conditions **X = "Sunny", "Hot", "Normal", "False"**

P(Y|X) & P(~Y|X) .... P(Y|X) = P(Y|S,H,N,F)

	Outlook	Temperature	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	High	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	Yes
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	No
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	Normal	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
False	6	2	6/9	2/5
True	3	3	3/9	3/5
Total	9	5	100%	100%

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Naive Bayes Classifier

Consider the problem of playing golf. Build a predictive model whether the day is suitable for playing (y), given conditions **X = "Sunny", "Hot", "Normal", "False"**

P(Y|X) = P(Y|S,H,N,F)

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

P(Y|X) = P(S|Y)P(H|Y)P(N|Y)P(F|Y)P(Y) / P(X)

Joint Probability = P(A ∩ B) = P(A) x P(B)

P(~Y|X) = P(S|~Y)P(H|~Y)P(N|~Y)P(F|~Y)P(~Y) / P(X)

** P(X) is common in both → ignore P(X)*

P(Y|X) = (2/9)*(2/9)*(6/9)*(6/9)*(9/14) = 0.0141

P(~Y|X) = (3/5)*(2/5)*(1/5)*(3/5)*(5/14) = 0.0068

Normalization: P(Y|X) = 0.0141/(0.0141+0.0068) = 0.67

P(~Y|X) = 0.0068/(0.0141+0.0068) = 0.33

P(Y|X) > P(~Y|X) → Game should be played today ('Yes')

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
False	6	2	6/9	2/5
True	3	3	3/9	3/5
Total	9	5	100%	100%

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Bayesian Inference

- In Statistics, there are 3 different approaches available to determine the probability of an event.
    - Classical, Frequentist, Bayesian
- Bayesian perspective allows to incorporate personal belief/opinion into the decision-making process.
- It takes into account what is already known about particular problem even before any empirical evidence.
- Bayesian inference techniques specify how one should update one's beliefs upon observing data.
- 4 main components of the Bayes' theorem: *Prior, Likelihood, Posterior, Evidence*.
- **Conjugate Priors** – Conjugacy occurs when the final posterior distribution belongs to the family of similar probability density functions as the prior belief but with new parameter values which have been updated to reflect new evidence/information.
- **Non-conjugate Priors**

Bayesian Decision Theory

- Bayesian Decision Theory is the statistical approach to pattern classification.
- It leverages probability to make classifications, and measures the risk (cost) of assigning an input to a given class.
  - Prior Probability, Likelihood Probability, Evidence
  - Sum of All Prior Probabilities Must be 1
  - Sum of All Posterior Probabilities Must be 1

$$P(C_1) + P(C_2) + P(C_3) + \ldots + P(C_K) = 1$$

$$P(C_1|X) + P(C_2|X) + P(C_3|X) + \ldots + P(C_K|X) = 1$$

P(w | x)= P(x | w) * P(w)P(x)

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

- **Prior – P(w)** is the Prior Probability that w is true before the data is observed
- **Posterior – P(w | x)** is the Posterior Probability that w is true after the data is observed.
- **Evidence – P(x)** is the Total Probability of the Data
- **Likelihood – P(x | w)** is the information about w provided by 'x'

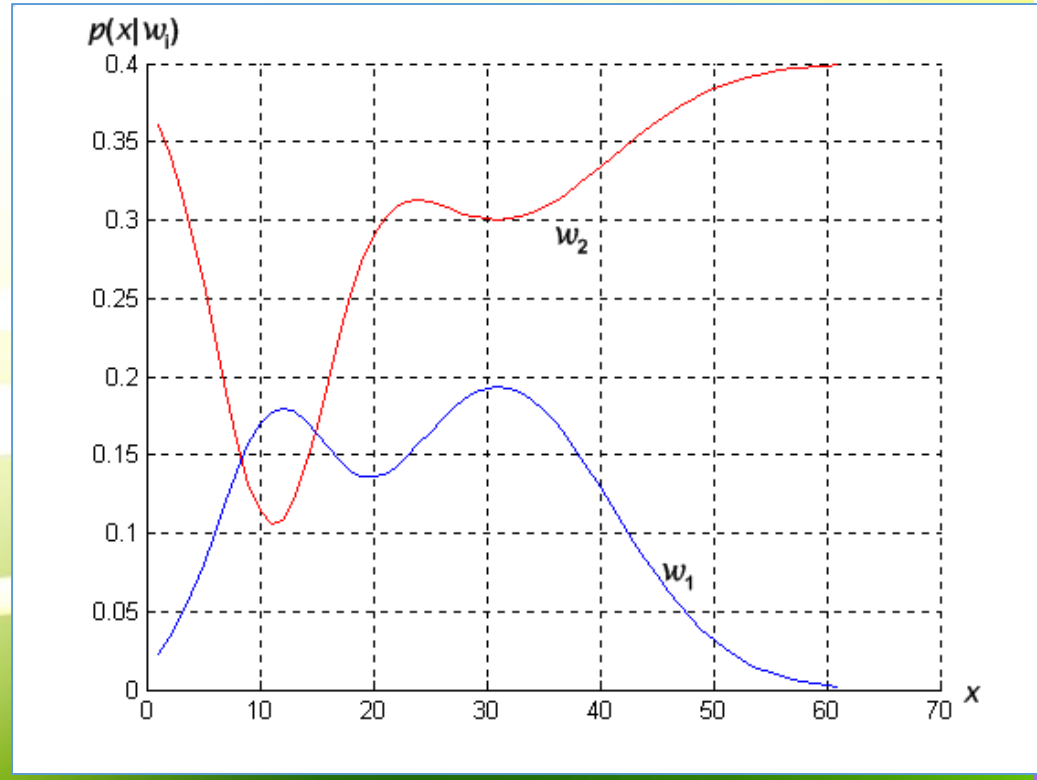
Bayesian Decision Theory

- If for an observation x, P(w1|x) > P(w2|x) → decide that the true state is w1.
- The probability of error is calculated as;

$$P(error \mid x) = \begin{cases} P(w_1 \mid x) & \text{if we decide } w_2 \\ P(w_2 \mid x) & \text{if we decide } w_1 \end{cases}$$

- Probability of total error for a feature x; P(E | x).
- Total error for a feature x would be;

P(E | x) = minimum (P(w1 | x) , P(w2 | x))



Bayesian Hypothesis Testing

- Suppose that we need to decide between two hypotheses H_0 and H_1.
- In the Bayesian setting, we assume that we know prior probabilities of H_0 and H_1.

  $P(H_0)=p_0$ & $P(H_1)=p_1$, where $p_0+p_1=1$.
- We know the distribution of Y (observed random variable) under the two hypotheses, i.e, we know $f_Y(y|H_0)$ & $f_Y(y|H_1)$.
- Using Bayes' rule, we can obtain the posterior probabilities of H_0 and H_1:

$$P(H_0|Y=y)=\frac{f_Y(y|H_0)P(H_0)}{f_Y(y)},$$
$$P(H_1|Y=y)=\frac{f_Y(y|H_1)P(H_1)}{f_Y(y)}.$$

- Compare $P(H_0|Y=y)$ & $P(H_1|Y=y)$ to accept hypothesis with higher posterior probability → maximum a posteriori (MAP) test.
- While choosing hypothesis with highest probability, it is relatively easy to show that the error probability is minimized.
- To be more specific, according to the MAP test, we choose H0 if and only if; $P(H_0|Y=y)\geq P(H_1|Y=y)$.