CM2104: Computational Mathematics
Discrete Probability Theory 1: Introduction

Prof. David Marshall

School of Computer Science & Informatics

Probability theory

- The first formal treatment of probability theory is due to Girolamo Cardano, who studied **games of chance** in his book *Liber de ludo aleae*
- Now it is a crucial component in many domains of computer science, e.g.
  - **Search engines** like Google are based on probabilistic language models to assess the relevance of a web page to a query, and on a random surfer model to assess the importance of a page
  - In **artificial intelligence**, probabilistic models are used to link features of objects to categories (machine learning), and to reason about actions with uncertain consequences (planning), among others
  - In computer networks, probability theory is used for **traffic modelling**.

Girolamo Cardano (1501-1576)

Probabilistic reasoning in AI

Association for Computing Machinery
Advancing Computing as a Science & Profession

site map | accessibility | contact

SEARCH

join | renew | subscribe | create web account

you are here: home → press room → current year news releases → 2012 → judea pearl wins acm a.m. turing award for contributions that transformed artificial intelligence

ACM | myACM

- Home
- Special Interest Groups (SIGs)
- Publications
- Membership
- Digital Library
- Libraries
- Educational Activities
- Learning Center
- Tech Packs
- Career & Job Center
- Chapters
- Conferences
- Calendar of Events
- Awards
- Boards & Committees
- Public Policy
- USACM
- ACM Blogs
- Computing News
- Buy Proceedings & Videos

Judea Pearl Wins ACM A.M. Turing Award for Contributions that Transformed Artificial Intelligence

Developed Novel Framework for Reasoning under Uncertainty that Changed How Scientists Approach Real World Problems

acm
The Association for Computing Machinery
Advancing Computing as a Science & Profession

Contact: Virginia Gold
212-626-0505
vgold@acm.org

Printable PDF file

NEW YORK, March 15, 2012—ACM, the Association for Computing Machinery today named Judea Pearl of the University of California, Los Angeles the winner of the 2011 ACM A.M. Turing Award for innovations that enabled remarkable advances in the partnership between humans and machines that is the foundation of Artificial Intelligence (AI). Pearl pioneered developments in probabilistic and causal reasoning and their application to a broad range of problems and challenges. He created a computational foundation for processing information under uncertainty, a core problem faced by intelligent systems. He also developed graphical methods and symbolic calculus that enable machines to reason about actions and observations, and to assess cause-effect relationships from empirical findings. His work serves as the standard method for handling uncertainty in computer systems, with applications ranging from medical diagnosis, homeland security and genetic counseling to natural language understanding and mapping gene expression data. His influence extends beyond artificial intelligence and even computer science, to human reasoning and the philosophy of science.

The Turing Award, widely considered the "Nobel Prize in Computing," carries a $250,000 prize, with financial support provided by Intel Corporation and Google Inc. It is named for the British mathematician Alan M. Turing, whose 100th anniversary will be celebrated in June at the ACM 2012 Turing Centenary Celebration that includes 34 past Turing Award winners along with Pearl.

Definitions

Probability theory deals with **(random) experiments**,
i.e. processes or actions whose outcome cannot be predicted with certainty, and may change when the experiment is repeated:

- Rolling a dice (possible outcomes: 1,...,6)
- Drawing two cards from a shuffled pack of cards (possible outcomes are pairs of different cards: $\{\clubsuit 5, \diamondsuit 6\}, \{\spadesuit H, \clubsuit A\}$, etc.)
- Flipping a coin (possible outcomes: *heads* and *tails*)

Probability theory aims at **quantifying** the **uncertainty** surrounding the possible **outcomes** of an experiment

Definitions

The **set** $S \neq \emptyset$ of all possible outcomes is called the **sample space**

A **probability distribution** is a **function** p from S to $[0, 1]$, such that

$$\sum_{s \in S} p(s) = 1$$

The **value** $p(s)$ is called the **probability** of seeing **outcome** s.

- It reflects how **likely** it is that the outcome of the associated experiment is s, **relative** to the other outcomes in S

In many cases, each of the **outcomes** in S is equally likely (*e.g.* Fair Dice, coin, Deck of Cards), in which case p is the **uniform distribution**, defined for all $s \in S$ by

$$p(s) = \frac{1}{|S|}$$

Definitions

A **subset** $A \subseteq S$ is called an **event**

With each **probability distribution** p, we can associate a **probability measure** P, as follows:

$$P : 2^S \to [0,1]$$
$$A \mapsto \sum_{a \in A} p(a)$$

For $A \in 2^S$, $P(A)$ is called the **probability** of **event** A.

- It reflects how **likely** it is that the outcome of the associated experiment will be among those in A.

Note that for the uniform distribution p, we have

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

Basic Properties of Probability

It is easy to see that

$$P(\emptyset) = 0 \qquad P(S) = 1 \qquad P(\overline{A}) = 1 - P(A)$$

where $P(\overline{A})$ (or (sometimes notated) $P(A^c)$ or $P(coA)$) is called the **complement** of A

If $A_1, ..., A_n$ are **pairwise disjoint** (*i.e.* $A_i \cap A_j = \emptyset$ for $i \neq j$), we have

$$P(A_1 \cup ... \cup A_n) = P(A_1) + ... + P(a_n)$$

Note that this implies for arbitrary events A and B that

$$\begin{aligned}
P(A \cup B) &= P((A \setminus B) \cup (A \cap B) \cup (B \setminus A)) \\
&= P(A \setminus B) + P(A \cap B) + P(B \setminus A) \\
&= (P(A \setminus B) + P(A \cap B)) + (P(B \setminus A) + P(A \cap B)) - P(A \cap B) \\
&= P((A \setminus B) \cup (A \cap B)) + P((B \setminus A) \cup (A \cap B)) - P(A \cap B) \\
&= P(A) + P(B) - P(A \cap B)
\end{aligned}$$

Examples

Rolling a single dice

Sample space $S = \{1, 2, 3, 4, 5, 6\}$

- The probability of rolling an even number: $P(\{2, 4, 6\}) = \frac{3}{6} = \frac{1}{2}$
- The probability of not rolling a 6: $P(\{1, 2, 3, 4, 5\}) = \frac{5}{6}$

Rolling two dice

Sample space $S = \{(i, j) \mid 1 \leq i, j \leq 6\}$

- The probability of rolling an even number:

$$P(sum\ even) = P(both\ even) + P(both\ odd) = \frac{3 \cdot 3}{36} + \frac{3 \cdot 3}{36} = \frac{1}{2}$$

- The probability of rolling a 7: $P(\{(i, j) \mid i + j = 7, 1 \leq i, j \leq 6\}) = \frac{6}{36} = \frac{1}{6}$

Note that for both experiments, the associated probability distribution p is uniform.

Counting

In many cases, **evaluating** the probability of an event amounts to **counting** the **number** of **outcomes** in the event and/or the **sample space**, *e.g.*:

- Evaluating the probability that you **win** the lottery **amounts** to **counting** how many ways there are to choose a **winning combination**, and how many **combinations** there are in **total**.
- Evaluating the probability that you roll a **9** using **two dice**, amounts to **counting** how **many** ways there are to get a **9**.
- Evaluating the probability that you are dealt a **full house**, when given **5 cards**, amounts to **counting** how many possibilities there are to make a **full house**, and how many **combinations** of hands there are in **total**.

Counting rules

Addition rule

Let $P_1, ..., P_k$ be **mutually exclusive properties** (i.e. no object can satisfy more than one of these properties), and let n_i be the number of objects that satisfies property P_i. Then the **number of objects** that satisfies **one** of the **properties** $P_1, ..., P_k$ is given by $n_1 + ... + n_k$

Example: Consider a box with 15 blue balls, 30 red balls and 10 yellow balls. There are **45 balls** we can choose from this box which are either **blue** or **red**.

Example: Knowing there are **two** ways to roll a 3 using two dice, and **three** ways to roll a 4, we find that there are **5 ways** to roll either a **3** or a **4**.

Subsets

Suppose the set X contains n objects, then there are 2^n **ways** of **choosing** a subset from X.

Example: A pizza company offers a basic pizza with a selection of 8 possible topics. The **total number** of **different** pizzas that can be chosen is given by 2^8.

Counting rules

Product rule

Suppose we have to **choose** a sequence$(x_1, ..., x_k)$ where there are n_1 options for the first element, for each choice of the first element, there are n_2 options for the second element, and in general, for each choice of$(x_1, ..., x_{i-1})$ there are n_i choices for the ith element. Then there are $n_1 \cdot ... \cdot n_k$ **ways** to **choose** a sequence

Example: Knowing there are **5 starters** to choose from, **4 main courses** to choose from, and **6 desserts**, we find that there are $5 \cdot 4 \cdot 6 = 120$ possible **three-course** menus to **choose** from.

Example: There are $3 \cdot 3 \cdot 3$ ways of rolling **only even numbers** when using **three** dice.

Note: Number of **subsets** of X, with$|X| = n$, can be derived from the **product rule**: there are **two choices** for the first element of X (include or don't include), **two** choices for the second element, ... Hence the total number of choices is $2 \cdot ... \cdot 2 = 2^n$.

Exercise

Poker: straight and straight flush

In the game of poker, a straight is a hand of 5 consecutive cards in which the 5 cards are not all of the same suit, e.g. $\{\clubsuit 5, \diamondsuit 6, \heartsuit 7, \clubsuit 8, \spadesuit 9\}$. A straight flush is a hand of 5 consecutive cards of the same suit, e.g. $\{\clubsuit 5, \clubsuit 6, \clubsuit 7, \clubsuit 8, \clubsuit 9\}$. A Royal flush is consecutive cards 10, Jack, Queen, King + Ace in same suit

1. How many ways are there to make a **royal** flush (assuming a standard deck of 52 cards)?
2. How many ways are there to make a **straight** flush ?
3. How many ways are there to make a **straight** (which is not a royal or straight flush)?

Ordering

Permutations

There are $n!$ ways of **ordering** a sequence of n **distinct** objects.

Indeed, there are n **ways** of choosing the first object, which leaves us with $n-1$ remaining options for the second option, $n-2$ options for the third option, *etc.* Using the product rule, we thus find that the total number of possibilities is $n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 1 = n!$

Example: In shuffle mode, an MP3 player with **10 songs** on it can choose from $10! = 3628800$ possible orderings

Example: There are $52! = 8.07x10^{67}$ ways in which a deck of cards may be ordered

MATLAB Permutation Examples

ordering.m

```
% In shuffle mode, an MP3 player with 10 songs on
% it can choose from 10! songs
n_song_choice = factorial(10)

% Let's prove it by emumeration
song_ID = 1:10; % 10 Unique Song IDs
song_perms = perms(1:10);

% Generate all Perms --- a lot;
num_perms = length(song_perms)

% There 52! ways to order a pack of cards
n_card_order = factorial(52)
```

Ordering

Distinguishable permutations

Assume you have n_1 **identical objects** of some kind, n_2 **identical objects** of a second kind, ... n_k **identical object** of the kth kind, with $n = n_1 + ... + n_k$.

Then the **number** of **distinguishable permutations** of the n **objects** is

$$\frac{n!}{n_1! \cdot n_2! \cdot ... \cdot n_k!}$$

There are $n!$ **possible** ways to choose the overall ordering.

However, there are $n_1!$ ways to reorder the positions of the objects of the first kind, $n_2!$ ways to reorder the positions of the objects of the second kind, *etc.*

By the **product rule**, that there are $n_1! \cdot n_2! \cdot ... \cdot n_k!$ ways to reorder any initial ordering, without actually changing it.

In other words, when counting $n!$ possible ways to choose the overall ordering, each possible distinguishable ordering is counted $n_1! \cdot n_2! \cdot ... \cdot n_k!$ times.

Ordering

Example: How many ways are there to reorder the letter tiles on the following Scrabble rack:

The different kinds here are the letters `'A'`, `'E'` and `'I'`, hence $n_1 = 1$, $n_2 = 5$, $n_3 = 1$, which means that the total number or distinguishable orderings is

$$\frac{7!}{1!5!1!} = \frac{7!}{5!} = 7 \cdot 6 = 42$$

If we care about the **relative position** of the different `'E'` tiles, on the other hand, there are 7! orderings

Note: The **permutation rule** can be derived from the above rule by considering that when all objects are **distinct**, we have $n = k$ and $n_1 = n_2 = ... = n_k = 1$ and thus $\frac{n!}{n_1! \cdot n_2! \cdot \ldots \cdot n_k!} = \frac{n!}{1! \cdot \ldots \cdot 1!} = n!$

MATLAB Code for Distinguishable Permutations

ordering.m

```
% How many ways can you reorder the letters of 'aeeeeei'
letter_perms = perms('aeeeeei');
num_perms = length(letter_perms)

% How many ways can you reorder the letters of 'aeeeeei' with
% distinguishable permutations
my_string = 'aeeeeei';

% Find unique characters
[unique_chars, unique_idx, char_idx] = unique(my_string);

% Count the occurence of each unique character
% Note: hist only deals with number so must use char_idx
count = hist(char_idx,unique(char_idx));

% number of distinguishable permutaions
dist_perms =  factorial(length(my_string))/prod(factorial(count))
```

Ordering

Recall that $X_1, ..., X_n$ is called a **partition** of a **set** Y **if** $X_1 \cup ... \cup X_n = Y$ and $X_i \cap X_j = \emptyset$ **for all** $i \neq j$.

The sets $X_1, ..., X_n$ are called the **partition classes**.

Counting the **number** of **partitions** with fixed sized partition classes is a special case of **counting distinguishable permutations**. In that case we view the objects as being fixed, and we reorder sequences of partition classes

Example: How many ways are there to divide a team of 15 rugby players into 3 front row players, 2 second row players, 3 back row players, 2 half-backs, 4 three-quarters, and a fullback?

This is equal to the number of ways in which we can order the sequence FFFSSBBBHHTTTTU, hence we arrive at $\frac{15!}{3! \cdot 2! \cdot 3! \cdot 2! \cdot 4!}$

MATLAB Code for Rugby Player Example

ordering.m

```
% How many ways are there to divide a team of 15 rugby players
% into 3 front row players, 2 second row players,
% 3 back row players, 2 half-backs,
% 4 three-quarters, and a fullback?

 my_string = 'FFFSSBBBHHTTTTU';

 % Find unique characters
[unique_chars, unique_idx, char_idx] = unique(my_string);

% Count the occurence of each unique character
% hist only deals with number so use char_idx
count = hist(char_idx,unique(char_idx));

% number of distinguishable permutaions
dist_perms =  factorial(length(my_string))/prod(factorial(count))
```

Exercise

MISSISSIPPI

How many different strings can you form by reordering the letters of the word MISSISSIPPI?

Group work

Eight students need to complete a group assignment. There are four students needed for programming, three students to do a requirement analysis, and one student to write the report. In how many ways can the eight students be assigned a task (programming / requirement analysis / report)?

See *ordering.m* for MATLAB code solution.

Combinations

Combinations

The **number of ways** k objects can be **selected** among n **distinct objects**, *regardless of the order*, is given by

$$^{n}C_{k} = \binom{n}{k} = \frac{n!}{k! \cdot (n-k)!}$$

Note: This amounts to **counting** the **number** of **ways to partition** a **set of** n **objects** in a **class of** k **objects** and a **class of** $n-k$ **objects**, so it can be viewed as a **special case** of **counting** the **number** of **distinguishable permutations**.

Combinations: Winning the National Lottery

Example: The **chance of winning** the national lottery (choosing **6** from **49**) was (**Pre Oct 2015**):

$$\frac{49!}{6!43!} = 13,983,816 \text{ to } 1.$$

or a probability of: $\frac{1}{13{,}983{,}816}$

The **current chances** of winning the national lottery (choosing **6** from **59**) has gone to:

$$\frac{59!}{6!53!} = 45,057,474 \text{ to } 1.$$

or a probability of: $\frac{1}{45{,}057{,}474}$

Combinations in MATLAB

MATLAB Combinations — nchoosek()

```
nchoosek Binomial coefficient or all combinations.
    nchoosek(N,K) where N and K are non-negative integers
    returns N!/K!(N-K)!.
    This is the number of combinations of N things taken K at a time.
    When a coefficient is large, a warning will be produced indicating
    possible inexact results. In such cases, the result is only accurate
    to 15 digits for double-precision inputs, or 8 digits for
    single-precision inputs.
```

Example

```
>>nchoosek(365,30)
Warning: Result may not be exact. Coefficient is greater than
9.007199e+15 and is only accurate to 15 digits
> In nchoosek (line 92)

ans =

   8.1848e+43
```

Note: You can turn MATLAB warnings **on**/**off** with command: `warning on`/`warning off`

Lottery Combinations MATLAB Code

ordering.m

```
% The chance of winning the national lottery

% Choosing 6 from 49 (Pre Oct 2015):
Lottery_Chance_2014 = nchoosek(49,6)
Lottery_Prob_2014 = 1/nchoosek(49,6)

% Choosing 6 from 59 (Post Oct 2015):
Lottery_Chance_2015 = nchoosek(59,6)
Lottery_Prob_2015 = 1/nchoosek(59,6)
```

Combinations

Example: In Texas hold 'em poker, each player is dealt **two "hole cards"** at the beginning.

The **number of combinations** of **hole** cards a player can receive is given by:

$$^{52}C_2 = \binom{52}{2} = \frac{52!}{2! \cdot 50!} = \frac{52 \cdot 51}{2} = 26 \cdot 51 = 1326$$

The **number of ways** in which a player can receive a **pair of threes**, for example, as **hole cards** is given by

$$^{4}C_2 = \binom{4}{2} = \frac{4!}{2! \cdot 2!} = \frac{4 \cdot 3 \cdot 2}{2 \cdot 2} = 6$$

The **probability** of being dealt a **pair** of **threes** is thus given by:

$$\frac{6}{1326} \approx 0.004524$$

The **probability** of being dealt **any pair** is given by:

$$13 \cdot \frac{6}{1326} = 0.05882$$

Texas Hold 'em MATLAB Code

ordering.m

```
% The number of combinations of hole cards a player can receive:
texas_hole_comb = nchoosek(52,2)

% Number of ways a player can recieve a pair of threes?
pair_three = nchoosek(4,2)

% Probability of pair of threes (for example):
prob_texas_pair_three = nchoosek(4,2)/nchoosek(52,2)

% Probability of Any pair  in Texas hold 'em poker
prob_texas_Any_pair = 13*nchoosek(4,2)/nchoosek(52,2)
```

Combinations

Birthday problem

What is the probability to find two people with the same birthday among a group of 30 people (ignoring leap years, and assuming that every day of the year is equally likely to be somebody's birthday)?

We make use of the fact that

$$P(\textit{at least two people have the same birthday}) = 1 - P(\textit{all people have a different birthday})$$

The total number of ways in which we can assign 30 different birthdays to 30 people is

$$30! \cdot {}^{365}C_{30} = 30! \cdot \binom{365}{30} = 30! \cdot \frac{365!}{30! \cdot 335!} = \frac{365!}{335!}$$

The total number of ways in which we choose 30 birthdays is 365^{30}. Hence the answer is given by

$$1 - P(\textit{all people have a different birthday}) = 1 - \frac{365!}{335! \cdot 365^{30}}$$

Birthday problem: MATLAB

ordering.m

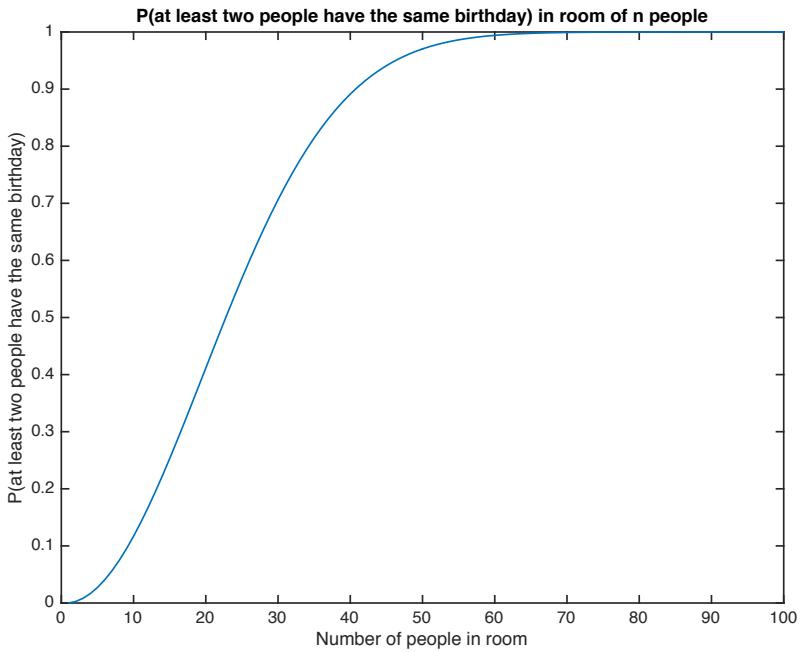
```
numofbirthdaysto30people = factorial(30)*nchoosek(365,30)

%The total number of ways in which we choose 30 birthdays is 365^30.
% Hence the answer is given by
P_same_Birthday = 1 - factorial(30).*nchoosek(365,30)./(365.^30)

% Note  Bad IDEA to Grow Your Own
(factorial(365))./(factorial(335).*(365.^30))

%%%ans = NaN!!
```

Birthday problem: 1:100 People in the Room



ordering.m

```
%%% Do this for n people n = 1:100
for i=1:100
    P_same_Birthday(i) = 1 - factorial(i).*nchoosek(365,i)./(365.^i);
end

plot(P_same_Birthday);
title('P(at least two people have the same birthday) in room of n people');
xlabel('Number of people in room');
ylabel('P(at least two people have the same birthday)');
shg;
```

Exercise

Poker: full house

In the game of poker, a full house is a hand consisting of 3 cards of the same kind and 2 cards of another kind, e.g. $\{\clubsuit 5, \diamondsuit 5, \heartsuit 5, \clubsuit 8, \spadesuit 8\}$. How many ways are there to make a full house from a standard deck of 52 cards?

Poker: double pair

In the game of poker, a double pair is a hand consisting of 2 cards of the same kind, 2 cards of another kind, and one card from yet another kind, e.g. $\{\clubsuit 5, \diamondsuit 5, \heartsuit 7, \clubsuit 7, \spadesuit 8\}$. How many ways are there to make a double pair from a standard deck of 52 cards?

See *ordering.m* for MATLAB code solution.

Independence

Two events A and B are called **independent** if

$$P(A \cap B) = P(A) \cdot P(B)$$

In such a case, the fact that A **occurs** **does not affect** the probability of B **occurring**, and **vice versa**.

Example

Sample space $S = \{1, 2, 3, 4\}$ with a uniform probability distribution, $A = \{1, 2\}$, $B = \{1, 3\}$. We find:

$$P(A) = P(B) = \frac{1}{2} \qquad P(A \cap B) = P(\{1\}) = \frac{1}{4}$$

Hence

$$P(A \cap B) = P(A) \cdot P(B)$$

so A and B are **independent**.

Conditional probability

In general, knowing that some event B **happened** (i.e. the outcome of the experiment under consideration is among those in the set B) **may affect** the **probability** that the event A has happened, **even if** we don't have any further information on the exact outcome of the experiment

We write $P(A|B)$ for the **conditional probability** of seeing A, given that B occurs

Consider the example of rolling a dice

- We don't know anything: $P(\{1,3\}) = \frac{1}{3}$
- We know that the outcome was **odd**: $P(\{1,3\}|\{1,3,5\}) = \frac{2}{3}$
- We know that the outcome was **even**: $P(\{1,3\}|\{2,4,6\}) = 0$
- We know that the outcome was **odd** and **at most 3:** $P(\{1,3\}|\{1,3\}) = 1$

Conditional probability

Definition

The **conditional probability** of A given B is defined as:

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

provided that $P(B) > 0$.

Note: If $P(B) = 0$, the conditional probability $P(A|B)$ is **not defined**; its value is then a matter of convention.

Note: From the definition of conditional probability, we can immediately derive the following rule for the **probability of an intersection**:

$$P(A \cap B) = P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

assuming $P(A) > 0$ and $P(B) > 0$

Bayes' theorem

Bayes' theorem

If $P(A) > 0$ and $P(B) > 0$, it holds that

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

Indeed, using the definition of conditional probability, we find that

$$P(A|B) \cdot P(B) = P(A \cap B) = P(B|A) \cdot P(A)$$

which immediately leads to the stated theorem

Example: Assume that 5% of incoming emails are spam. Moreover, assume that 50% of spam emails contain the word *account*. In contrast, only 10% of all emails contain the word *account*. What is the probability that an email is spam, given that it contains the word *account*?

$$P(spam|account) = \frac{P(account|spam) \cdot P(spam)}{P(account)} = \frac{0.5 \cdot 0.05}{0.1} = 0.25$$

Partition theorem

Partition theorem

If $B_1 \cup \ldots \cup B_k$ forms a partition of the sample space S, it holds that

$$P(A) = \sum_{i=1}^{k} P(A|B_i) \cdot P(B_i)$$

Indeed, since the sets $A \cap B_1$, ..., $A \cap B_k$ are all pairwise disjoint, and

$$A = A \cap S = A \cap (B_1 \cup \ldots \cup B_k) = (A \cap B_1) \cup \ldots \cup (A \cap B_k)$$

we find

$$P(A) = P((A \cap B_1) \cup \ldots \cup (A \cap B_k)) = P(A \cap B_1) + \ldots + P(A \cap B_k)$$

from which we find the stated theorem

Note: This theorem is also known as the **law of total probability**.

Partition theorem

Example: Consider a dice which has a probability of $\frac{3}{4}$ of being fair, and a probability of $\frac{1}{4}$ of being biased such that

$$p(1) = p(2) = p(3) = p(4) = p(5) = \frac{1}{10} \qquad p(6) = \frac{1}{2}$$

What is the probability of rolling an even number?

Using the partition theorem, we find

$$\begin{aligned} P(\{2,4,6\}) &= P(\{2,4,6\}|\text{fair}) \cdot P(\text{fair}) + P(\{2,4,6\}|\text{biased}) \cdot P(\text{biased}) \\ &= \frac{1}{2} \cdot \frac{3}{4} + \frac{1+1+5}{10} \cdot \frac{1}{4} \\ &= \frac{3}{8} + \frac{7}{40} \\ &= \frac{22}{40} \\ &= \frac{11}{20} \end{aligned}$$

Bayes' Theorem with the Partition theorem

We may rewrite **Bayes' Theorem** using the Partition theorem as:

$$P(A_i|B) = \frac{P(B|A_i) \cdot P(A_i)}{P(B)}$$
$$= \frac{P(B|A_i)P(A_i)}{\sum_{k=1}^{n} P(B|A_k)P(A_k)}$$

Exercise

Given **three** cards:

- One with **red** on **each side**
- One with **green** on **each side**
- One with **red** on **one side** and **green** on the **other**.

If we draw **one** card at **random** and look at **one side only**:

- What is the **probability** that given the side we are looking at is **green** that the **other side** is **also green**?

Exercise

A box contains 3 **yellow balls** and 5 **red balls**.
All are chosen at **random** from the box, then **replaced** in the box along with **two other balls** of the **same colour**.

1. If a **second ball** is now chosen at random from the box, what is the **probability** that it will be **red**?
2. Given that the **second ball** is **red**, what is the **probability** that the first ball was **yellow**?