Overview

- More counting
- Permutations
- Permutations of repeated distinct objects
- Combinations
More counting

We’ll need to count the following again and again, let’s do it once:

• Counting the number of ways to generate an ordered subset of size $k$ from a set of $n$ distinguishable objects (Permutation)

• Counting the number of ways to generate an unordered subset of size $k$ from a set of $n$ distinguishable objects (Combination)

We’ll see that counting permutations is based on the product rule of counting and counting combinations is based on permutations.
More counting

Examples:

- Counting the number of distinct pizzas we can create by selecting 4 toppings from 6 available.
- How many distinct lottery numbers when choose 6 in range 1-47.
- How many ways can 3 bit errors occur in a string of 8 bits.
- How many ways can I allocate 50 servers from a pool of 100 servers.
- How many routes are there between two points in a network.
Permutations

**Permutation**: An ordered arrangement

- Example. How many ways can we arrange the letters in the word “abc”? 

```
  a  b  c
  b  a  c
  c  b  a
  a  c  b
  b  c  a
  c  a  b
```

The first letter can be chosen from any of 3, the second from any of 2, the third from 1. So by the product rule there are $3 \times 2 \times 1 = 6$ possible permutations

- Recall $n! = n(n - 1)(n - 2) \cdots 3 \times 2 \times 1$. So $3! = 3 \times 2 \times 1 = 6$

- In general, number of permutations of $n$ objects in $n!$ – by direct application of product rule.
Permutations

How many ways can we arrange the letters in the word “moo”? 

• Label the letters uniquely \( mo_1 o_2 \). Then we have \( 3! = 6 \) permutations, same as “abc”.

\[
\begin{array}{ccc}
m & o_1 & o_2 \\
m & o_2 & o_1 \\
o_1 & m & o_2 \\
o_2 & m & o_1 \\
o_1 & o_2 & m \\
o_2 & o_1 & m \\
\end{array}
\]

• But if treat the two o’s as the same we get only 3 distinct arrangements:

\[
\begin{array}{ccc}
m & o & o \\
o & m & o \\
o & o & m \\
\end{array}
\]

• Take \( mo_2 o_1 \), If we permute the o’s we get \( mo_1 o_2 \) but it still reads moo. There are \( 2! = 2 \) ways to permute the two o’s. So we need to divide 3! by 2!, which gives us \( 6/2 = 3 \) permutations
Permutations

A slightly harder example: “pepper”.

• Three p’s, two e’s and one r. Label as \( p_1 e_1 p_2 p_3 e_2 r \).

• Consider one permutation e.g. ppeper. How many equivalent ways can we write this?

\[
\begin{array}{cccccc}
  p_1 & p_2 & e_1 & p_3 & e_2 & r \\
  p_1 & p_3 & e_1 & p_2 & e_2 & r \\
  p_2 & p_1 & e_1 & p_3 & e_2 & r \\
  p_2 & p_3 & e_1 & p_1 & e_2 & r \\
  p_3 & p_1 & e_1 & p_2 & e_2 & r \\
  p_3 & p_2 & e_1 & p_1 & e_2 & r \\
\end{array}
\]

• Can arrange \( p_1, p_2, p_3 \) in 3! different orders. Can arrange \( e_1, e_2 \) in 2! different orders. Can arrange \( r \) in 1! = 1 different ways (trivially)

• So 3!2!1! = 12 ways to write ppeper.

• 6! ways to arrange \( p_1 e_1 p_2 e_2 p_3 e_2 r \). So \( \frac{6!}{3!2!1!} = 60 \) possible letter arrangements.
Permutations

With permutations of repeated distinct objects in general we have the following. Permuting \( n \) objects with \( k \) groups (first group has \( n_1 \) objects, second \( n_2 \) objects etc):

- Consider all of the \( n \) objects to be distinct at first and compute \( n! \)
- For the first distinct group with \( n_1 \) objects, divide \( n! \) by the permutations of this group \( n_1! \). Repeat for the second group with \( n_2 \) objects, and so on.
- Number of permutations is

\[
\frac{n!}{n_1!n_2! \cdots n_k!}
\]

- In the special case when \( k = n, \ n_1 = 1 = n_2 = \cdots = n_n \) then we get back to \( \frac{n!}{1!} = n! \).
Combinations

Interested in counting the number of different groups of $k$ objects that can be formed from a total of $n$ objects. Now order does not matter.

- Example: How many groups of 3 letters could be selected from the set of 5 letters $\{A, B, C, D, E\}$?

- There are 5 ways to select the first letter, 4 ways to select the second letter, 3 ways to select the third letter. So $5 \times 4 \times 3 = 60$ ways of selecting a group when the order matters.

- What about when the order doesn't matter?

- Each group containing letters $A, B, C$ is counted in the 60. There are 6 such groups: $ABC, ACB, BAC, BCA, CAB$ and $CBA$. Lumping these together we need to divide 60 by 6 to get number of groups when don't care about letter order.

- Frame it as a repeated permutation problem ... for each group of 3 letters there are $3! = 3 \times 2 \times 1 = 6$ permutations, so number of unordered groups is $\frac{5 \times 4 \times 3}{3 \times 2 \times 1}$.
Combinations

• In general there are \(n(n-1)(n-2) \cdots (n-k+1)\) ways that a group of \(k\) items can be selected from \(n\) items, when order matters.

• Each group of \(k\) items will be counted \(k!\) times in this count, so we need to divide by this to get number of unordered groups. That is, number of different groups of \(k\) objects that can be formed from a total of \(n\) objects is

\[
\frac{n(n-1)(n-2) \cdots (n-k+1)}{k!} = \frac{n!}{(n-k)!k!}
\]

• Notation: for \(0 \leq k \leq n\) define \(\binom{n}{k}\) by

\[
\binom{n}{k} = \frac{n!}{(n-k)!k!}
\]

We say that \(\binom{n}{k}\) is number of possible combinations of \(n\) objects taken \(k\) at a time. Say “\(n\) choose \(k\)”.

• Note that \(0! = 1\) by convention. So \(\binom{n}{0} = 1\) and \(\binom{n}{n} = 1\).
Combinations

Example:

• How many ways can 3 bit errors occur in a string of 8 bits. \( \binom{8}{3} = 56 \).

• How many ways can I allocate 50 servers from a pool of 100 servers. \( \binom{100}{50} \approx 10^{29} \).

• Number of distinct pizzas we can create by selecting 4 toppings from 6 available. \( \binom{6}{4} = 15 \).

• How many distinct lottery numbers when choose 6 in range 1-47. \( \binom{47}{6} = 10,737,573 \)
Combinations

Pizza toppings:
- Gorgonzola
- Olives
- Peppers
- Mushrooms
- Artichokes
- Epoisses de Bourgogne\(^1\)

How many different combinations? \(\binom{6}{4} = 15\). But can’t use Gorgonzola and Epoisses together as just too stinky. How many different combinations now?

\(^1\)Apparently banned from public transport in Paris, Napoleon’s favourite
Combinations

Solution 1:

- Case 1: Gorgonzola and 3 other toppings (excluding Epoisses). $\binom{4}{3}$
- Case 2: Epoisses and 3 other toppings (excluding Gorgonzola). $\binom{4}{3}$
- Case 3: 4 toppings that aren’t Gorgonzola or Epoisses. $\binom{4}{4}$
- Total is $\binom{4}{3} + \binom{4}{3} + \binom{4}{4} = 9$

Solution 2:

- All combinations. $\binom{6}{4}$
- Gorgonzola + Epoisses + 2 other toppings. $\binom{4}{2}$
- Remainder: $\binom{6}{4} - \binom{4}{2} = 9$
**Power Sets**

- **Power set of** $S$: the set of all subsets of $S$, including the empty set and $S$ itself. Sometimes written $2^S$.

- Example: $S = \{A, B, C\}$,

  \[2^S = \{\emptyset, \{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}, \{A, B, C\}\}\]

- Note that in a set the elements are unordered i.e. set $\{A, B\}$ is same as set $\{B, A\}$

  \[|2^S| = \binom{3}{0} + \binom{3}{1} + \binom{3}{2} + \binom{3}{3} = 1 + 3 + 3 + 1 = 8\]
Power Sets

• Let $|S| = n$. In general,

$$|2^S| = \sum_{k=0}^{n} \binom{n}{k}$$

• **Binomial Theorem**: $(x + y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$ (see book for proof)

• Example:

$$|2^S| = \sum_{k=0}^{n} \binom{n}{k} = \sum_{k=0}^{n} \binom{n}{k} 1^k 1^{n-k} = (1 + 1)^n = 2^n$$

• So $|2^S| = 2^{|S|} = 2^n$
Basket Data

- Basket data also called transaction data.
- Plenty of it.
- Example:

<table>
<thead>
<tr>
<th>ID</th>
<th>apples</th>
<th>beer</th>
<th>cheese</th>
<th>eggs</th>
<th>ice cream</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
Basket Data

<table>
<thead>
<tr>
<th>ID</th>
<th>apples</th>
<th>beer</th>
<th>cheese</th>
<th>eggs</th>
<th>ice cream</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Discovering “rules”.

- A rule is something like this: *If a basket contains beer then it also contains ice cream*
- Accuracy: when the if part is true, how often is the then part true.
- Coverage: how much of the database contains the if part
- 5 out of 8 entries contain beer (coverage is $\frac{5}{8} = 0.625$). Of these 3 also contain ice cream (accuracy is $\frac{3}{5} = 0.6$).
- Is this rule interesting/surprising i.e. do beer and ice cream appear in same basket more than we would expect by chance?
Basket Data

<table>
<thead>
<tr>
<th>ID</th>
<th>apples</th>
<th>beer</th>
<th>cheese</th>
<th>eggs</th>
<th>ice cream</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- \( \frac{5}{8} = 0.625 \) of baskets contain beer, \( \frac{3}{8} = 0.375 \) contain ice cream. So if these are independent and we pick a basket uniformly at random we expect \( 0.625 \times 0.375 \approx 0.23 \) of baskets to contain both.
- Is observed fraction 0.6 with beer and ice cream interestingly larger than 0.23?
- Depends on the amount of data (only 8 baskets, but what if had 1M baskets? Or 100M?). Depends on our assumptions e.g. independence.
- For large data sets, can’t enumerate all possible “rules”. Smart algorithms for enumerating rules with specified minimum coverage, see https://en.wikipedia.org/wiki/Apriori_algorithm.
Prediction: Regression

We have some data e.g. scores in ST3009 tutorials and in final exam:

\[
\begin{array}{cccccc}
3 & 7 & 2 & 9 & 1 & 75 \\
5 & 8 & 2 & 9 & 2 & 85 \\
4 & 1 & 1 & 1 & 3 & 25 \\
6 & 8 & 2 & 1 & 4 & 55 \\
\end{array}
\]

We get some new data:

\[
\begin{array}{cccccc}
3 & 6 & 1 & 8 & 1 & ? \\
\end{array}
\]

- Can we accurately predict the final exam score with high probability?
- E.g. picking a number between 0 and 100 uniformly at random is certainly a prediction, but hopefully a poor one.
- Expect that quality of prediction depends on the amount of data and on our assumptions.
Prediction: Classification

We have some data which is labelled $A$ or $B$ e.g. has passed ST3009 exam:

```
3 7 2 9 1 A
5 8 2 9 2 A
4 1 1 1 3 B
6 8 2 1 4 A
```

We get some new data:

```
3 6 1 8 1 ?
```

- Can we accurately predict the label $A$ or $B$ with high probability?