# Counting

Carlo Tomasi

To count means to determine the cardinality of a set, that is, the number of elements it contains. The cardinality of a set $S$ is denoted by $\sharp(S)$.

This note shows two different counting methodologies. The first one establishes a calculus for the cardinalities of sets constructed from sets of known cardinalities through the following composition operators: (i) union of disjoint sets ("set addition"); (ii) set difference between a set and a subset of it ("set subtraction"); (iii) Cartesian product of sets. In contrast with other set operators such as union and intersection, the cardinalities of the sets obtained through these three operators are known functions of the cardinalities of the constituent sets.

The second method can be applied whenever a set can be defined by specifying a procedure for constructing its elements. The cardinality of the set in question is then determined by counting the options available for constructing a general member of the set.

## 1 Counting through Set Composition

Sets are often built by applying composition operators such as union, intersection, complement, Cartesian product, and so forth to other sets. It would be convenient if were possible to express the cardinality of such compositions as a function of the cardinalities of the constituent sets. This is possible for the cross product (or Cartesian product) of sets:

$$\sharp(A \times B) = \sharp(A)\sharp(B) \; . \tag{1}$$

When cross products are computed, the universes $U_A$ and $U_B$ in which $A$ and $B$ are defined are multiplied as well to yield $U = U_A \times U_B$ for $A \times B$.

Unfortunately, no counting rules are generally associated with the other set-composition operators. For instance, the cardinality of the union or intersection of two sets $A$ and $B$ cannot be determined from those of $A$ and $B$. We only have bounds:

$$\max(\sharp(A), \sharp(B)) \leq \sharp(A \cup B) \leq \sharp(A) + \sharp(B)$$

where the lower bound is attained when one set is contained in the other, and the upper bound is attained when the two sets are disjoint. Similarly,

$$0 \leq \sharp(A \cap B) \leq \min(\sharp(A), \sharp(B)) \; .$$

Here, the lower bound is attained when the two sets are disjoint and the upper bound when one set is contained in the other.

When $A$ and $B$ are not disjoint, the following *inclusion-exclusion formula* holds:

$$\sharp(A \cup B) = \sharp(A) + \sharp(B) - \sharp(A \cap B)$$

because the elements in the intersection are counted once in $\sharp(A)$ and once again in $\sharp(B)$. Similar results can be obtained by *partitioning* the union of $A$ and $B$, that is, by rewriting it as a union of disjoint sets:

$$A \cup B = A \cup (\overline{A} \cap B) = (A \cap \overline{B}) \cup B = (A \cap \overline{B}) \cup (\overline{A} \cap B) \cup (A \cap B) \ .$$

This then yields the following three *partitioned counting* formulas:

$$\sharp(A \cup B) = \sharp(A) + \sharp(\overline{A} \cap B) = \sharp(A \cap \overline{B}) + \sharp(B) = \sharp(A \cap \overline{B}) + \sharp(\overline{A} \cap B) + \sharp(A \cap B) \ . \qquad (2)$$

These results are useful when the cardinalities of the various intersections that appear in them can be computed in some separate way. Fortunately, this is often the case.

**Exercise.** Draw Venn diagrams for all the formulas above.

A useful formal device for counting is to use a different symbol for the union of two sets when the two sets are disjoint:

$$A + B = \begin{cases} A \cup B & \text{if } A \cap B = \emptyset \\ \text{undefined} & \text{otherwise} \end{cases}$$

and to say that in that case $A$ and $B$ are *added* together. Set addition should not be viewed as a new set operator, but merely as the old union operator with emphasis added on the fact that $A$ and $B$ are disjoint. It is an error to write $A + B$ if $A$ and $B$ are not disjoint, just as it would be an error to write $'abc' + 2$ without having defined the meaning of '+' when applied to a string and a number.

If $C = A + B$, then we can write $A = C - B$ or $B = C - A$ as long as we interpret the set *subtraction* operator '$-$' as follows:

$$C - B = \begin{cases} C \setminus B & \text{if } B \subseteq C \\ \text{undefined} & \text{otherwise} \end{cases}$$

where '$\setminus$' is the set difference. Note in particular that if $U$ is the universe, then the complement of a set $A$ can be written as $\overline{A} = U - A$. This is useful whenever computing the cardinality of $A$ is hard but computing those of $U$ and $\overline{A}$ is easier, or vice versa.

In summary, the elements of a set $S$ can be counted as follows:

- Determine a collection of *basic sets* $S_1, \ldots, S_n$ whose cardinality is known.
- Write $S$ as a combination of cross products and set addition and subtraction operations, starting from the basic sets.
- In the resulting combination, replace the basic sets with their cardinalities, set additions with sums, set subtractions with differences, and set cross products with products.
- Compute the value of the resulting formula.

## 2 Example

Consider a simplified password scheme for some security application. Valid characters for a password are the 26 letters of the alphabet (case insensitive, so 'A' and 'a' are the same letter) and the ten digits from 0 to 9. A password is a nonempty string of up to three characters, at least one of which must be a digit. If more

digits are present, they must be different. For instance, the strings '2' '15', '317', 'A6C', '2B' are valid. Examples of invalid passwords are 'C11' (digit repetition), 'ABC' (no digit), and 'BC36' (too long). How many passwords are possible in this scheme?

In this case, basic sets are simple to define: The universe $C$ for each character in the password is the set of alphanumeric characters, and $\sharp(C) = 36$. Digits and letters form the sets $D$ and $L$ with cardinality 10 and 26, respectively.

The set $P$ of passwords can be written as the set addition of the sets of valid passwords with one, two, or three characters:

$$P = P_1 + P_2 + P_3$$

where

$$\begin{aligned} P_1 &= \{\text{passwords with one character}\} \\ P_2 &= \{\text{passwords with two characters}\} \\ P_2 &= \{\text{passwords with three characters}\} \ . \end{aligned}$$

These sets are disjoint, because a password cannot have two different lengths at the same time. A one-character password must be a digit, so we easily have

$$P_1 = D \ ,$$

a basic set. $P_2$ can be written as the set $Q_2$ of all two-character passwords that ignore the rule about distinct digits, minus the set $V_2$ of those that violate it (a subset of $Q_2$). Violations are most easily listed:

$$V_2 = \{00, 11, 22, 33, 44, 55, 66, 77, 88, 99\} \ ,$$

so this can be considered a basic set. The set $Q_2$ can be specified in different ways. One is to first start with a union

$$\begin{aligned} Q_2 &= \{\text{two-character passwords starting with a digit}\} \\ &\cup \{\text{two-character passwords ending with a digit}\} \ . \end{aligned}$$

[Note that the first component of $Q_2$ contains all the repeated-digit pairs in $V_2$.] These two sets are not disjoint, so they can be partitioned:

$$\begin{aligned} Q_2 &= \{\text{two-character passwords starting with a digit}\} \\ &+ \{\text{two-character passwords starting with a letter and ending with a digit}\} \end{aligned}$$

(without this partitioning, we would be counting passwords with two digits twice) and we can immediately write these two sets as cross products of basic sets:

$$Q_2 = (D \times C) + (L \times D) \ .$$

In summary,

$$P_2 = Q_2 - V_2 = ((D \times C) + (L \times D)) - V_2 \ .$$

Handling $P_3$ takes more work. First, we can write

$$P_3 = Q_3 - V_3$$