Writing in Computer Science: Math

Javier Larrosa

UPC Barcelona Tech

.

Introduction

In Technical and Scientific documents mathematical expressions are mixed with English.

In this lecture we will start seeing how Math and English should be mixed.

Definitions

In Scientific Writing, we often have to write **definitions**.

- Definitions in words may be easier to understand, but may lack precision
- Definitions in Math may be more precise, but may be hard to understand
- Some concepts are more suitable for one or the other.

So what to do?

- Definitions can be given in **words** or in **mathematical writing**
- Sometimes we give both
- Sometimes we add an example

Example: mean, median, mode

in words

Given a list of numerical values,

- The **mean** is the sum of the values divided by the number of values
- The **median** is the "middle" value after sorting the list. If the number of values is even, the median is the mean of the two "middle" values.
- The **mode** is the value that occurs most often. In case of ties, there is no mode.

Mean (math writing)

Mean

Given n numerical values $x_1, x_2, \ldots, x_n$, the **mean** is

$$\frac{\sum_{i=1}^{n} x_i}{n}$$

Median (math writing)

Median

Given n **sorted** numerical values $x_1, x_2, \ldots, x_n$, the **median** is,

- $x_{(n/2)+1}$, if n is odd
- $\frac{x_{n/2}+x_{(n/2)+1}}{2}$, if n is even

Mode (math writing)

Mode

Consider a list of numerical values $L = (x_1, x_2, \ldots, x_n)$, possibly with repetitions. Let $Y = \{y_1, y_2, \ldots, y_m\}$ the set of distinct values appearing in L. Let f_i (with $1 \leq i \leq m$) be the number of occurrences of y_i in L,

$$f_i = |\{j,\ 1 \leq j \leq n |\ x_j = y_i\}|$$

The **mode** of L is y_{i^*} where i^* is defined as,

$$i^* = \mathtt{argmax}_{i=1}^{m}\{f_i\}$$

Some hints from Donald Knuth

On notation revision

When you first establish notation you may not be fully aware of what are you going to need from it. So be ready (and willing) to make changes

Modularity

Modularity

When having to write a complex definition decompose the definition into simpler concepts and build the definition in a modular way
Finding the right modularity is far from trivial

As we do when we write algorithms....

Some hints from Donald Knuth

- Is a bad idea to start with a definition like "let $X = \{x_1, x_2, \ldots, x_n\}$" if you are going to need subsets of X. Also you will need to be speaking of elements x_i and x_j all the time.
- Do not name the elements of a set X unless necessary. Thus you can alway refer to elements x and y of X, and a subset Y of X.

Combinatorial Auction

Supose I have a collection of stamps that I want to sell. Some stamps are new and others are used. Some stamps are from Spain, others from other countries in Europe or the rest of the world. The stamps are from different time periods.

- One bidder offers some money for all the Spanish stamps
- One bidder offers money for all new new stamps
- One bidder offers money for all the *XIX* century stamps
- ...

and I want to make as much profit as possible from my collection.

Example: Auction (first version)

Given a set of items S, a **bid** is a pair (B, v) with $B \subset S$ and v being a positive number. An **auction** $\mathcal{A}$ is a set of bids.
Let $\mathcal{S}$ be a subset of $\mathcal{A}$. We say that $\mathcal{S}$ is **feasible** if for all pair of bids (B, v) and (B', v') in $\mathcal{S}$, we have that $B \cap B' = \emptyset$.
The **value** of $\mathcal{S}$ is,

$$f(\mathcal{S}) = \sum_{(B,v) \in \mathcal{S}} v$$

The goal is to find,

$$\max_{\mathcal{S} \text{ feasible}} \{f(\mathcal{S})\}$$

Example: Auction (first version MADE A BIT MORE GENTLE)

Given a set of items S, a **bid** is a pair (B, v) with $B \subset S$ and v being a positive number. ELEMENT B REPRESENTS THE ITEMS REQUESTED BY THE BID AND v IS THE OFFERING PRICE.
An **auction** $\mathcal{A}$ is a set of bids. Let $\mathcal{S}$ be a subset of $\mathcal{A}$. We say that $\mathcal{S}$ is **feasible** IF THERE ARE NO ITEMS APPEARING IN MORE THAN ONE OF ITS BIDS. Formally, $B \cap B' = \emptyset$, for all pair of bids (B, v) and (B', v') in $\mathcal{S}$,
The **value** of $\mathcal{S}$, noted $f(\mathcal{S})$, IS THE SUM OF ITS OFFERING PRICES,

$$f(\mathcal{S}) = \sum_{(B,v)\in\mathcal{S}} v$$

THE GOAL IS TO FIND THE FEASIBLE SUBSET OF A WITH THE HIGHEST VALUE.

Example: Auction (second version)

Given a set of items S, a **bid** is a pair $b = (B, v)$ with $B \subset S$ and v being a positive number. Element B represents the items requested by the bid and v is the offering price.

An **auction** is a set of bids $b_1, b_2, \ldots, b_n$, with $b_i = (B_i, v_i)$. Let N be a subset of $\{1, 2, \ldots, n\}$ which represents a subset of the auction. We say that N is **feasible** if there are no items appearing in more than one of its bids,

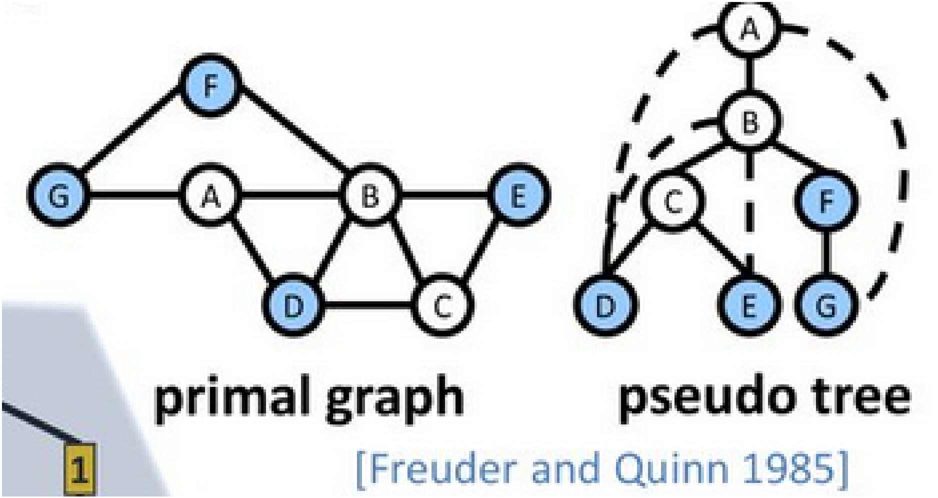
$$\forall_{i,j \in N, i \neq j} B_i \cap B_j = \emptyset$$

The **value** of N, noted $f(N)$, is the sum of its offering prices,

$$f(N) = \sum_{i \in N} v_i$$

The goal is to find the feasible set N with the highest value.

Example: Pseudo-tree



Example: Pseudo-tree (first version)

A **tree** is a cycle-free graph.... The **ancestors** of a vertex u are all the vertices in the **path** from u to the tree **root**.

Definition

We say that a tree $T = (V', E')$ with root $r \in V'$ is a **pseudo-tree** of a graph $G = (V, E)$ if: both have the same set of vertices (e.g. $V = V'$) and every edge in G does not cross branches in T (e.g. for all $(u, v) \in E$ we have that $u \in anc(T, v)$ or $v \in anc(T, u)$

Obs: I do not like the prime appearing before the non-prime. Think about fixing it.
Obs: I do not like the fact of introducing the root but never using it. (root is important, though, to give meaning to branch)

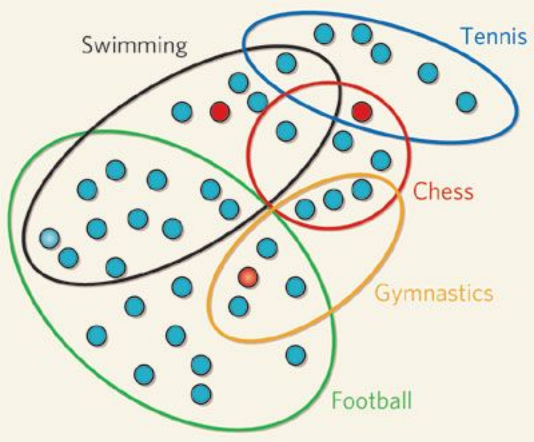
Example: Pseudo-tree, second version

A **tree** is a..., a **rooted tree** is..., the **ancestors** of a vertex u are all the vertices in the **path** from u to the tree **root**.

Definition

A pseudo-tree of a graph $G = (V, E)$ is a partition of the edges $A \cup B = E$ such that $T = (V, A)$ is a tree with root $r \in A$ and every edge $(u, v) \in B$ does not cross branches in T (e.g. for all $(u, v) \in B$ we have that $u \in anc(T, v)$ or $v \in anc(T, u)$)

Example: Hitting Set



Example: Hitting Set

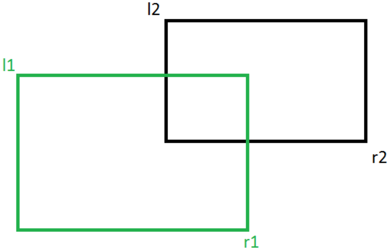
Hitting Set

Let $S_1, S_2, \ldots, S_n$ be subsets of a set U. We say that a set $H \subseteq U$ is a **hitting set** if every S_i has at least one element in H,

$$\forall_{1 \leq i \leq n}, |S_i \cap H| > 0$$

In the **hitting set problem** the goal is to find a hitting set of minimum size.

Example: Non-overlapping Rectangles



Example: Non-overlapping Rectangles

We represent a rectangle as a pair of two-dimension points $r = ((x_1, y_1), (x_2, y_2))$ where the first point indicates the top-left corner and the second point indicates the bottom-right corner.
Two rectangles $r = ((x_1, y_1), (x_2, y_2))$ and $r' = ((x'_1, y'_1), (x'_2, y'_2))$ **do not overlap** iff

- $(x_2 < x'_1)$ (that is, r is at the left of r') or,
- $(x'_2 < x_1)$ (that is, r is at the right of r') or,
- $(y_1 < y'_2)$ (that is, r is below r') or,
- $(y'1 < y_2)$ (that is, r is above r')