Dynamic Programming

Arijit Bishnu
arijit@isical.ac.in

Indian Statistical Institute, India.

August 31, 2015

Outline

Outline

Maximum sum contiguous subsequence

Let S be an array, such that $S[i] \in \mathbb{R}$. We want to find a contiguous subsequence of S with the maximum sum. The problem can be solved with the following recurrence.

$$S[i] = \max\{S[i-1] + a[i], a[i]\} \forall i, 1 \leq i \leq n-1$$

Exercise

Find the indices k and ℓ, $1 \leq k \leq \ell \leq n$, such that $\sum_{i=k}^{\ell} S[i]$ is maximum.

Outline

Longest common subsequence (LCS)

Let $A = \{a_1, a_2, \ldots, a_n\}$ and $B = \{b_1, b_2, \ldots, b_m\}$ be two strings over an alphabet set Σ. A subsequence of A is of the form $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$, where each i_j is between 1 and n and $1 \leq i_1 < i_2 < \cdots < i_k \leq n$. Let $L[i,j]$ denote the length of the LCS of $\{a_1, a_2, \ldots, a_i\}$ and $\{b_1, b_2, \ldots, b_j\}$. Then, $L[i,j]$ can be expressed as

$$L[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0; \\ L[i-1, j-1] + 1 & \text{if } a_i = b_j,\ i,j > 0 \\ \max\{L[i-1,j], L[i,j-1]\} & \text{if } a_i \neq b_j,\ i,j > 0 \end{cases}$$

The final answer is $L[n,m]$.

Exercise

Find the subsequences of A and B that achieve the LCS.

Outline

Matrix chain multiplication

We have to find the matrix product $\mathcal{M} = M_1 \times M_2 \times \cdots \times M_k \times M_{k+1} \times M_{k+2} \times \cdots M_n$, where M_k is of size $r_k \times r_{k+1}$.

Let $C[i, j]$ be the minimum cost of multiplying the matrices $M_i \times \cdots \times M_j$ in terms of scalar multiplications. We denote $M_i \times \cdots \times M_j$ as $M_{i,j}$. We can write $M_{ij} = M_{i,k-1} \times M_{k,j}$ for $i < k < j$.

$$C[i, j] = \min_{i \leq k \leq j} \{C[i, k-1] + C[k, j] + r_i r_k r_{j+1}\}$$

$C[1, n]$ is the final answer.

Exercise

Find out how to compute $C[i, j]$.

Outline

All pair shortest path

Let $G = (V, E)$ be a directed graph with weights on each edge. Let $V = \{1, \ldots, n\}$. We want to find the distance among all vertices of G. Let $i, j \in V$ and d_{ij}^k be the length of the shortest path from i to j that does not pass through $\{k+1, k+2, \ldots, n\}$. So, d_{ij}^n is the final answer.

$$d_{ij}^k = \begin{cases} \ell[i,j] & \text{if } k = 0 \\ \min\left\{d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1}\right\} & \text{if } 1 \leq k \leq n \end{cases}$$

Outline

Knapsack

Let $U = \{u_1, \ldots, u_n\}$ be a set of n items to be packed inside a knapsack of capacity C, $C \in \mathbb{Z}^+$. For $1 \leq i \leq n$, let s_i and v_i be the size and value of the i-th item, where $s_i, v_i \in \mathbb{Z}^+$. We want to fill the knapsack with some items from U whose total size is at most C and the sum of the values of the items in the knapsack is maximized. Let $V[i, j]$ denote the value obtained by filling a knapsack of size j with items taken from the first i items, $\{u_1, \ldots, u_i\}$ in an optimal way.

$$V[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ V[i-1,j] & \text{if } j < s_i \\ \max\{V[i-1,j], V[i-1,j-s_i] + v_i\} & \text{if } i > 0 \text{ and } j \geq s_i \end{cases}$$

$V[n, C]$ is the final answer.

Exercise

Find the subset of U that gives the optimal answer.