Miguel Angel Ruiz Ortiz

Dynamic Programming + Li Chao Tree

Problema: Sea $A[1,\ldots,N]$ un arreglo de N enteros, $1 \leq A[i] \leq 10^6$, $1 \leq N \leq 10^4$. Definimos el costo del subarreglo $A[l,\ldots,r]$ como $(A[r]-A[l])^2$. Dado $1 \leq K \leq \min\{N, 100\}$, debemos particionar el arreglo en K subarreglos no vacíos tal que el costo total de los subarreglos sea mínimo, es decir, la suma de los costos de cada subarreglos sea mínima, e imprimir dicho costo total.

Solución: Consideremos la siguiente $DP[\,][\,]$ de dos estados:

$$DP[m][i] = \text{el costo total mínimo de partir } A[1,\ldots,i] \text{ en } m \text{ subarreglos, } m \leq i.$$

Notemos $DP[1][i] = (A[1]-A[i])^2$ para todo i y $DP[m][m] = 0$. A continuación veremos la transición de la DP.

Al particionar el arreglo $A[1,\ldots,i]$, tenemos que escoger en índice k en el que empezará el m-ésimo subarreglo, y después partir el subarreglo $A[1,\ldots,k-1]$ en $m-1$ subarreglos. Como debe haber al menos $m-1$ elementos en $A[1\ldots,k-1]$, entonces $m \leq k \leq i$. Así, el costo total mínimo de partir $A[1,\ldots,i]$ en m subarreglos es escoger el k que minimiza $(A[k]-A[i])^2$ el costo de $A[k,\ldots,i]$ más el costo total mínimo de partir $A[1,\ldots,k-1]$ en $m-1$ subarreglos. Es decir,

$$DP[m][i] = \min_{m\leq k\leq i}\{DP[m-1][k-1] + (A[k]-A[i])^2\}. \quad (\star)$$

Para calcular el mínimo en $(\star)$, usaremos el Li Chao Tree. Supongamos $c_{m,k} = DP[m-1][k-1]$ ya está calculado para $m \leq k \leq i$ y consideremos el conjunto de funciones $\{f_{m,k}\}_{m\leq k\leq i}$ dadas por

$$f_{m,k}(x) = c_{m,k} + (A[k]-x)^2 = c_{m,k} + A[k]^2 - 2A[k]x + x^2.$$

Estas funciones se intersectan a lo más una vez ya que

$$\begin{aligned} c_{m,k} + A[k]^2 - 2A[k]x + x^2 = f_{m,k}(x) &= f_{m,k'}(x) = c_{m,k'} + A[k']^2 - 2A[k']x + x^2 \\ \Leftrightarrow \qquad 0 &= 2x(A[k']-A[k]) + A[k]^2 - A[k']^2 + c_{m,k} - c_{m,k'}, \end{aligned}$$

y esta última ecuación tiene a lo más una solución para x. De modo que podemos encontrar el mínimo de estas funciones evaluadas en $x = A[i]$ con el Li Chao-Tree en $\log N$, y entonces encontraremos

$$DP[m][i] = \min_{m\leq k\leq i}\{f_{m,k}(A[i])\}$$

de manera óptima.

Iremos llenando la matriz $DP[\,][\,]$ por filas. Para llenar la fila m necesitamos de la fila $m-1$. La primer fila ya mencionamos cómo se llena. También mencionamos $DP[m][m] = 0$. Supongamos ya tenemos calculado $DP[m][i]$ y tenemos un Li Chao Tree con las funciones $f_{m,k},\ k = m,\ldots,i$. Para calcular $DP[m][i+1]$, simplemente agregamos la función $f_{m,i+1}$ al Li Chao Tree que teníamos anteriormente en $\log N$ y buscamos la función que minimiza la evaluación en $A[i+1]$ también en $\log N$. De esta manera podemos ver que llenar la matriz $DP[\,][\,]$ tendrá complejidad $O(KN\log N)$.

□