CSE515 Advanced Algorithms
Lecture 4: Dynamic Programming II

Antoine Vigneron

antoine@unist.ac.kr

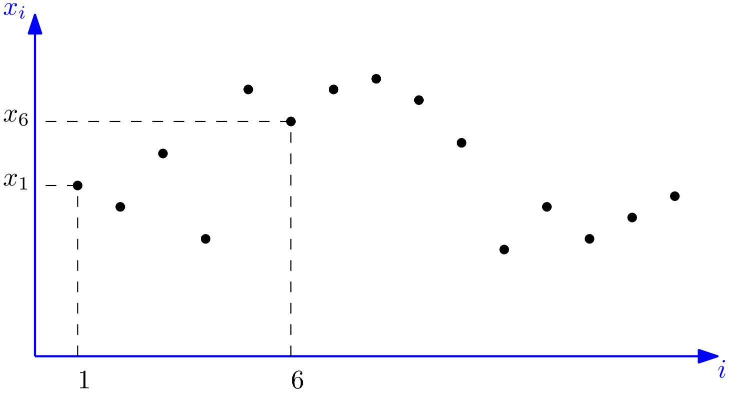
Ulsan National Institute of Science and Technology

March 11, 2021

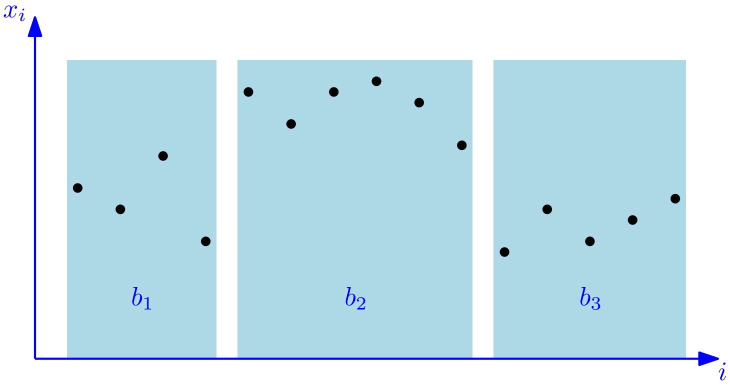
Introduction

- This is the second lecture on *Dynamic programming*.
- In the first lecture, we presented Dynamic Time Warping (DTW).
- In this lecture, we consider a historgram construction problem

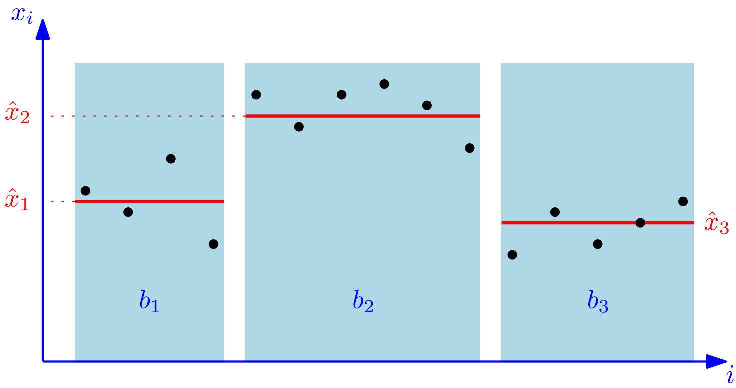
Histogram Construction



Histogram Construction



Histogram Construction



Problem Statement

- INPUT: integer m, sequence $x_1, \ldots, x_n$
- We will partition it into m *buckets* $b_1, \ldots, b_m$ of consecutive indices such that:
- $\forall p,\ b_p = \{s_p, s_p + 1, \ldots, e_p\}$,
- $\forall p < m,\ e_p = s_{p+1} - 1$
- $s_1 = 1,\ e_m = n$

Example

In the previous slide,
$s_1 = 1,\ e_1 = 4,\ s_2 = 5,\ e_2 = 10,\ s_3 = 11,\ e_3 = 14$

Problem Statement

- Each bucket b_p is represented by a value $\hat{x}_p$.
- The goal is to minimize $\sum_{p=1}^{m} f(s_p, e_p, \hat{x}_p)$ where

$$f(s_p, e_p, \hat{x}_p) = \sum_{i \in b_p} (\hat{x}_p - x_i)^2$$

over all choices of b_p and $\hat{x}_p$, $1 \leqslant p \leqslant m$.

Motivation

- Databases:
  - Summarizing a large dataset using a small histogram that fits in RAM
  - Answering approximate queries
  - Query optimization
- Computer graphics:
  - Curve simplification

Computing x_p^*

Lemma

Let $|b_p|$ denote the number of elements in b_p, and thus $|b_p| = |e_p - s_p + 1|$. Then the optimal value of $\hat{x}_p$ for this bucket is

$$x_p^* = \frac{1}{|b_p|} \sum_{i \in b_p} x_i.$$

Proof.

Done in class. □

Corollary

We denote $f^*(s_p, e_p) = f(s_p, e_p, x_p^*)$. Then we can compute x_p^* and $f^*(s_p, e_p)$ in $O(|b_p|)$ time.

Brute Force Approach

- Brute force:
  - Check all possible ways of partitioning into m buckets.
  - For each bucket b_p, compute x_p^* and $f^*(s_p, e_p)$.

Proposition

- *There are* $\binom{n-1}{m-1}$ *ways of choosing the* m *buckets.*
- $\binom{n}{m} = \Theta(n^m)$ *when* m *is a fixed constant.*
- $\binom{n}{\lfloor n/2 \rfloor} = \Omega(2^{n/2})$

- Conclusion:
  - Running time $\Omega(n\binom{n-1}{m-1})$.
  - For arbitrary m, this is exponential.
  - Only doable for small values of m such as $m = 3$ or 4.

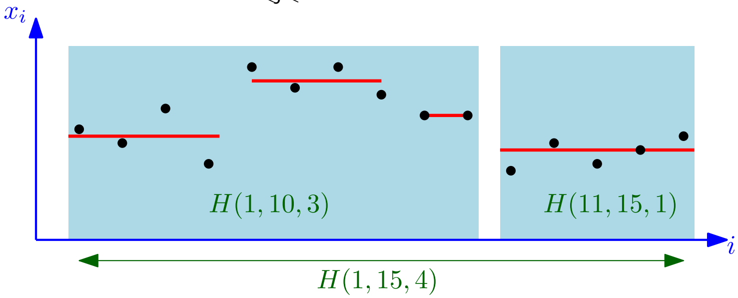
Dynamic Programming Approach

Subproblems

Let $H(i, j, k)$ be the value of an optimal k-buckets histograms for the subsequence $(x_i, x_{i+1}, \ldots, x_j)$.

- Recurrence relation: $\forall 2 \leqslant k \leqslant i \leqslant n$

$$H(1, i, k) = \min_{k-1 \leqslant j \leqslant i-1} \Big(H(1, j, k-1) + H(j+1, i, 1) \Big)$$



Pseudocode

Algorithm 3

function HISTOGRAM$(n, x_1, \ldots, x_n, m)$
  if $m \geqslant n$ **then**
    return 0
  $H \leftarrow$ new $n \times n \times m$ array
  for $i \leftarrow 1, n$ **do**
    for $j \leftarrow i, n$ **do**
      $H[i, j, 1] \leftarrow f^*(i, j)$
  for $k = 2, \ldots, m$ **do**
    for $i \leftarrow k, n$ **do**
      $H[1, i, k] \leftarrow \min_{k-1 \leqslant j \leqslant i-1} \Big(H[1, j, k-1] + H[j+1, i, 1] \Big)$
  return $H[1, n, m]$

Analysis

Proposition

Algorithm 3 computes the cost of an optimal histogram in $O(n^3)$ time.

Proof.

Line 7 takes $O(n)$ time, so the nested loops 5–7 take $O(n^3)$.
Line 10 takes $O(n)$ time, so the nested loops 8–10 take $O(mn^2)$.
We may assume that $m < n$, as otherwise the solution is trivial. $\square$

- There are at least 3 issues with Algorithm 3. Can you see them?

Concluding Remarks

- The running time and space usage of Algorithm 3 can be improved to $O(mn^2)$ and $O(mn)$, respectively. See lecture notes.
- No better algorithm is currently known.
- If we change the objective function, then better algorithms may exist.
- For instance, if one wants to minimize the maximum error, in other words, we want to minimize

$$\max_{p \in \{1,\ldots,m\}} \max_{i \in b_p} |\hat{x}_p - x_i|$$

then an $O(n)$-time algorithm is known.
(Fitting a Step Function to a Point Set. H. Fournier and A. Vigneron, Algorithmica 60: 95-109, 2011.)