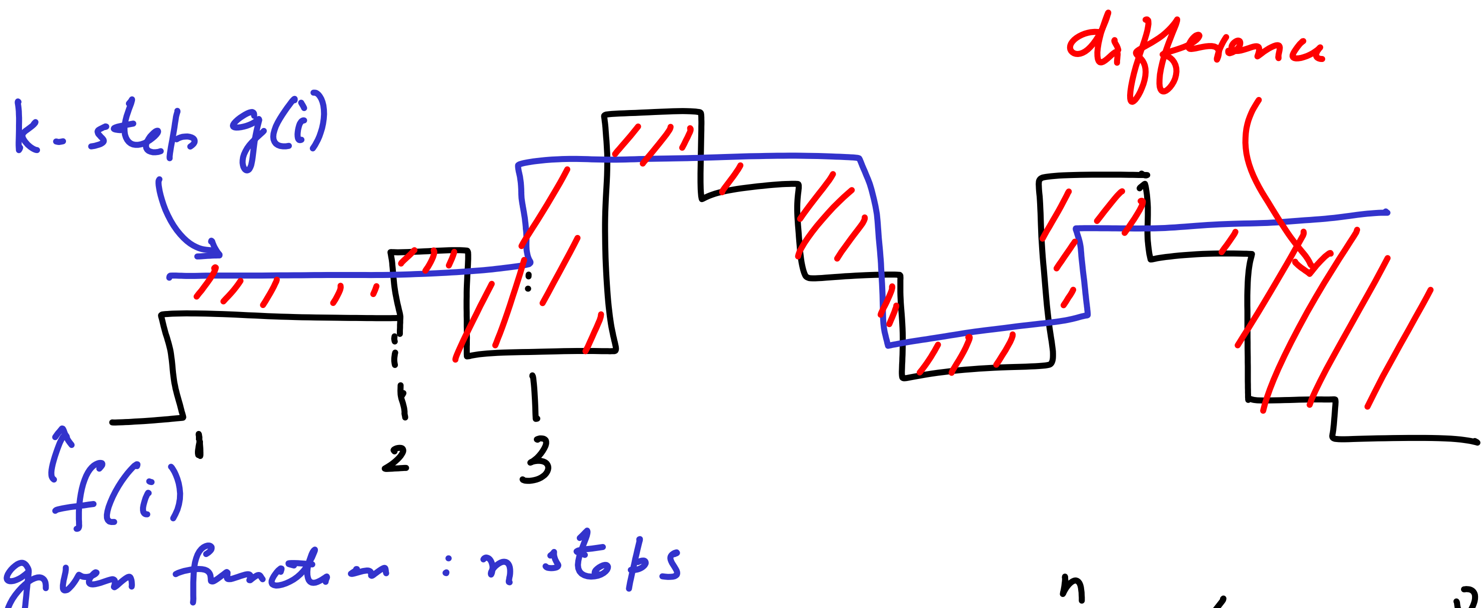
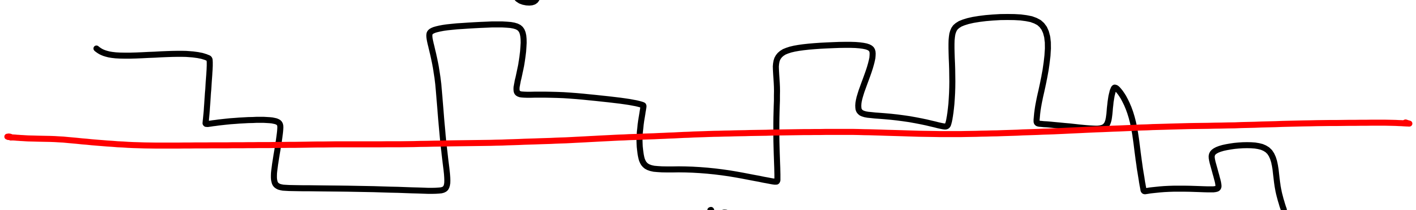
Lecture 17 Sept 22 CS 630

Function approximation



Minimize sum of squares $\sum_{i=1}^{n} \left(g(i) - f(i)\right)^2$

Special case: $g(i)$ is constant



Minimized when $g = \frac{1}{n}\left(\sum_{i=1}^{n} f(i) \right)$

Prove this

<u>Obs</u>: For the optimal g^*, the value at the k^{th} step = average of the f values greater than i_k (?)

g is defined at i_1 i_2, i_3 .. i_k where $i_1, i_2 \ldots i_k \in \{1 \ldots n\}$

If we knew i_k, we can define $g^*(i_k)$

Try all possibilities $i_k \in \{1 \ldots n\}$

$\{n-k \ldots n\}$ will suffice

For each value of i_k, we need an optimal g^* to approximate f between $1 .. i_k$

$g^*_{i,j}$ is the optimal i step function that approximates $f(1)\text{-}f(2) .. f(j)$

We are interested in $g^*_{k,n}$

$g^*_{1,j}$: $\frac{1}{j} \sum_{i=1}^{j} f(i)$ base case

Time : ?

$g^*_{i+1,j}$: Assume that we have computed $g^*_{i,j}$ for all $\boxed{i \le j \le n}$

Find the optimal $(i+1)^{st}$ step for $g^*_{i+1,j}$ using all possibilities between i and j and choose the best

Let $t(i,j)$ be the optimal $i+1^{st}$ step

Then $g^*_{i+1,j}$: $g^*_{i,t(i,j)}$ average of values between $t(i,j)$ and j

$j \le n$

we have from prior computation

try for $j-i$ terms

In increasing i and for a fixed $i \le k$ in increasing $j \le n$

A (i,j) is the avg of values from i to j $j > i$

If we compute $A(i,j)$ in advance the running time is $\sum_{i=1}^{k} \sum_{j=i}^{n} (j-i)$

Total time. + computing all $A(i,j)$

$1 \le i \le j \le n$

$$\le \sum_{i=1}^{k} \sum_{j=i}^{n} j = O(kn^2) +$$

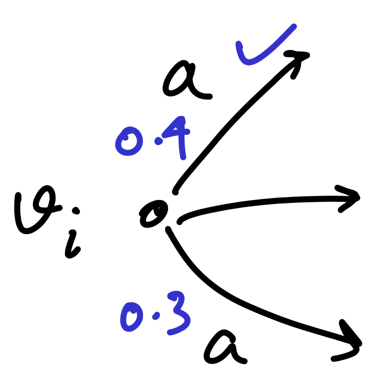
An edge labelled graph on some finite alphabet with $|V|$ vertices and $|E|$ edges.
Each edge also has a prob associated with it.

We want to find a path starting from v_0 with labels $\sigma_1 \sigma_2 \cdots \sigma_n$, $\sigma_i \in$ alphabet

s.t. $v_0 \xrightarrow{\sigma_1} v_1 \xrightarrow{\sigma_2} v_2 \cdots \xrightarrow{\sigma_n} v_n$

such that $p(v_0, v_1) \times p(v_1, v_2) \cdots p(v_{n-1}, v_n)$ is maximized.

$v_i \in V$ $\qquad$ $p(v_i, v_{i+1})$ is a prob measure



If the string was length 1, we choose the edge with the correct label and maxm prob

If we can solve the problem for lengths upto l then for length $l+1$ we can apply induction