Lecture: Analysis of Algorithms (CS483 - 001)

Amarda Shehu

Spring 2017
Probabilistic Analysis

- Average Case Analysis of Insertion Sort
**Definition**

Let $T(n)$ denote the average case time complexity used by an algorithm to solve a problem on an input size $n$. Then:

$$T(n) = \sum_{I \in D_n} P(I) \circ t(I)$$

- $D_n$ is the set of all input instances of size $n$
- $I$ denotes instance $I$ taking values over sample space $D_n$
- $P(I)$ denotes the probability with which $I$ occurs
- $t(I)$ denotes time it takes to solve problem on input instance $I$
- $\sum_{I \in D_n} P(I) = 1$ for correct analysis
Light Exercise: Average Case Analysis of Insertion Sort
Need a bit of a refresher on expected values and random variables.
Need a bit of a refresher on *expected* values and *random variables*
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$$

*Expected number of H’s from one flip of a fair coin is 1/2.*
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$$

*Expected number of H’s from one flip of a fair coin is 1/2.*

Q: What is the expected number of Heads in $n$ tosses of a coin?
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$$

*Expected number of H’s from one flip of a fair coin is 1/2.*

Q: What is the expected number of Heads in $n$ tosses of a coin?

Let $X = \sum_{i=1}^{n} X_{H,i}$ be the total number of H’s in $n$ tosses.
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$$

Expected number of H’s from one flip of a fair coin is $1/2$.

Q: What is the expected number of Heads in $n$ tosses of a coin?

Let $X = \sum_{i=1}^{n} X_{H,i}$ be the total number of H’s in $n$ tosses.

Then: 

$$E[X] = E[\sum_{i=1}^{n} X_{H,i}] = \sum_{i=1}^{n} E[X_H] = \sum_{i=1}^{n} 1/2 = n/2$$
Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable $X_H$ to track this number

$$E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2$$

*Expected number of H’s from one flip of a fair coin is $1/2$.*

Q: What is the expected number of Heads in $n$ tosses of a coin?

Let $X = \sum_{i=1}^{n} X_{H,i}$ be the total number of H’s in $n$ tosses.

Then:

$$E[X] = E[\sum_{i=1}^{n} X_{H,i}] = \sum_{i=1}^{n} E[X_{H,i}] = \sum_{i=1}^{n} 1/2 = n/2$$

*Expected number of H’s from $n$ tosses of a fair coin is $1/2$.***
Refresher in Context of Simple Coin Tossing Example

Q: What is the expected number of Heads from one coin toss?

Introduce binary random variable \( X_H \) to track this number

\[
E[X_H] = 1 \cdot P(X_H = 1) + 0 \cdot P(X_H = 0) = 1 \cdot (1/2) + 0 \cdot (1/2) = 1/2
\]

*Expected number of H’s from one flip of a fair coin is 1/2.*

Q: What is the expected number of Heads in \( n \) tosses of a coin?

Let \( X = \sum_{i=1}^{n} X_{H,i} \) be the total number of H’s in \( n \) tosses.

Then:
\[
E[X] = E[\sum_{i=1}^{n} X_{H,i}] = \sum_{i=1}^{n} E[X_H]
\]

\[
= \sum_{i=1}^{n} 1/2 = n/2
\]

*Expected number of H’s from \( n \) tosses of a fair coin is 1/2.*
**InsertionSort**\( (\text{array} A[1 \ldots n]) \)

1: for \( j \leftarrow 2 \) to \( n \) do
2: Temp \( \leftarrow A[j] \)
3: \( i \leftarrow j - 1 \)
4: while \( i > 0 \) and \( A[i] > \) Temp do
5: \( A[i + 1] \leftarrow A[i] \)
6: \( i \leftarrow i - 1 \)
7: \( A[i + 1] \leftarrow \) Temp

- Loop invariant: At the start of each iteration \( j \), \( A[1 \ldots j - 1] \) is sorted.

Recall:
\[
T(n) = \sum_{j=2}^{n} \{ A + \sum_{i=0}^{j-1} B + C \}
\]

Ignoring machine-dependent constants, we can write:
\[
T(n) = \sum_{j=2}^{n} k_j, \text{ where } k_j \text{ is a variable that tracks the total number of iterations of the inner while loop in an iteration of the outer for loop}
\]

In the worst-case analysis, we assumed that \( k_j \leq j \), arriving at a total quadratic running time for insertion sort.

*Here we ask for \( E[k_j] \)*
Average Case Analysis of Insertion Sort

\( k_j \): random variable counting total number of moves to the right

So: \( E[k_j] = E[\sum_{i=1}^{j-1} k_i] \), where \( k_i \) is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: \( E[k_j] = \sum_{i=1}^{j-1} E[k_i] \)

What is \( E[k_i] \)?
Average Case Analysis of Insertion Sort

\( k_j \): random variable counting total number of moves to the right

So: \( E[k_j] = E[\sum_{i=1}^{j-1} k_i] \), where \( k_i \) is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: \( E[k_j] = \sum_{i=1}^{j-1} E[k_i] \)

What is \( E[k_i] \)?

\[ E[k_i] = P(\text{move}) \times 1 + P(\text{no move}) \times 0 \]
Average Case Analysis of Insertion Sort

\( k_j \): random variable counting total number of moves to the right

So: \( E[k_j] = E[\sum_{i=1}^{j-1} k_i] \), where \( k_i \) is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: \( E[k_j] = \sum_{i=1}^{j-1} E[k_i] \)

What is \( E[k_i]？ \)

\[ E[k_i] = P(\text{move}) \times 1 + P(\text{no move}) \times 0 \]

\[ P(\text{move}) = P(A[i] > \text{Key}) = 0.5 \]
Average Case Analysis of Insertion Sort

$k_j$: random variable counting total number of moves to the right

So: $E[k_j] = E[\sum_{i=1}^{j-1} k_i]$, where $k_i$ is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: $E[k_j] = \sum_{i=1}^{j-1} E[k_i]

What is $E[k_i]$?

$E[k_i] = P(move) \times 1 + P(no\ move) \times 0$

$P(move) = P(A[i] > Key) = 0.5$

So: $E[k_i] = 0.5 \times 1 = 0.5 \implies E[k_j] = \sum_{i=1}^{j-1} 0.5 = \frac{j-1}{2}$

Finally:

$E[T(n)] = \sum_{j=2}^{n} j - 1 \cdot 0.5 = \frac{j-1}{2}$

You can show that this expected running time is no better than the worst-case running time.
**Average Case Analysis of Insertion Sort**

- **$k_j$:** random variable counting total number of moves to the right

So: $E[k_j] = E[\sum_{i=1}^{j-1} k_i]$, where $k_i$ is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: $E[k_j] = \sum_{i=1}^{j-1} E[k_i]$

What is $E[k_i]$?  

$E[k_i] = P(move) \times 1 + P(no \ move) \times 0$

$P(move) = P(A[i] > Key) = 0.5$

So: $E[k_i] = 0.5 \times 1 = 0.5 \Rightarrow E[k_j] = \sum_{i=1}^{j-1} 0.5 = \frac{j-1}{2}$

Finally: $E[T(n)] = \sum_{j=2}^{n} \frac{j-1}{2}$
k_j: random variable counting total number of moves to the right

So: \( E[k_j] = E[\sum_{i=1}^{j-1} k_i] \), where \( k_i \) is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: \( E[k_j] = \sum_{i=1}^{j-1} E[k_i] \)

What is \( E[k_i] \)?

\( E[k_i] = P(move) \times 1 + P(no \ move) \times 0 \)

\( P(move) = P(A[i] > Key) = 0.5 \)

So: \( E[k_i] = 0.5 \times 1 = 0.5 \implies E[k_j] = \sum_{i=1}^{j-1} 0.5 = \frac{j-1}{2} \)

Finally: \( E[T(n)] = \sum_{j=2}^{n} \frac{j-1}{2} \)

You can show that this expected running time is no better than the worst-case running time.
Average Case Analysis of Insertion Sort

$k_j$: random variable counting total number of moves to the right

So: $E[k_j] = E[\sum_{i=1}^{j-1} k_i]$, where $k_i$ is a random variable tracking the number of moves in one iteration of the while loop

By linearity of expectation: $E[k_j] = \sum_{i=1}^{j-1} E[k_i]$

What is $E[k_i]$? $E[k_i] = P(move) \times 1 + P(no \ move) \times 0$

$P(move) = P(A[i] > Key) = 0.5$

So: $E[k_i] = 0.5 \times 1 = 0.5 \implies E[k_j] = \sum_{i=1}^{j-1} 0.5 = \frac{j-1}{2}$

Finally: $E[T(n)] = \sum_{j=2}^{n} \frac{j-1}{2}$

You can show that this expected running time is no better than the worst-case running time.
Can we do better than $\theta(n^2)$?

You have have already seen an example ...
Can we do better than $\theta(n^2)$?

You have already seen an example ... 

More follow
Can we do better than $\theta(n^2)$?

You have already seen an example ...

More follow