

1 Bucket Sort

1.1 Pseudocode

Note: Assume you have an uniformly distributed array of real numbers between 0 and 1. Create n buckets.

Algorithm 1 Bucket Sort

```
1: for  $i = 0$  to  $n - 1$  do  $B[i] \leftarrow 0$  ▷ Clear  $B$   
  
2: for  $i = 1$  to  $n$  do  
3:    $B[\lfloor n \cdot A[i] \rfloor] \leftarrow A[i]$   
4: end for  
  
5: Sort  $B$  ▷ Can be done with any sorting algorithm  
6: Concatenate  $B$ 
```

1.2 Analysis

On average there will be one item per bucket, and lucky for us it is unlikely that a bucket will have a lot of elements in it. So average case, bucket sort is $O(n)$ or *linear* time.

1.3 Modifications

Between Two Real Numbers

What if we want to sort numbers between $[q, r)$? We can simply renormalize B like so.

```
1: for  $i = 0$  to  $n - 1$  do  $B[i] \leftarrow 0$  ▷ Clear  $B$   
  
2: for  $i = 1$  to  $n$  do  
3:    $B\left[\left\lfloor \frac{n \cdot A[i] - q}{r - q} \right\rfloor\right] \leftarrow A[i]$   
4: end for  
  
5: Sort  $B$  ▷ Can be done with any sorting algorithm  
6: Concatenate  $B$ 
```

Nonuniform Distribution

What if the data is not uniformly distributed? We can set the buckets' spread larger and larger as it expands out from the mean.

