CS325 Homework 1

Kabir Kang, Paul Ely, Jason Dorweiler

April 24, 2014

1 Section 1

```
if len(array) == 1:
     maxSum = array[0]
else:
     for e in range(len(array)):
          for j in range(e,len(array)):
               maxSum = np.maximum(maxSum, sum(array[e:j]))
```

Listing 1: pseudo code for n^3 algorithm

```
for e in range(len(array)):
     testSum = 0
     for j in range(e,len(array)):
          testSum += array[j]
          maxSum = np.maximum(maxSum, testSum)
```

Listing 2: pseudo code for n^2 algorithm

```
def algo3(array):
if(len(array) == 0):
     return 0
if(len(array) == 1):
     return array[0]

mid = len(array)/2
tempL = tempR = 0
maxLeft = maxRight = -99999

#left side crossing -- mid backwards
for i in range(mid,0,-1):
     tempL = tempL + array[i]
     maxLeft = np.maximum(maxLeft, tempL)

#right side crossing -- mid forwards
for j in range(mid+1, len(array)):
     tempR = tempR + array[j]
     maxRight = np.maximum(maxRight, tempR)
maxCrossing = maxLeft + maxRight

MaxA = algo3(array[:mid])
MaxB = algo3(array[mid+1:])

return np.maximum(np.maximum(MaxA, MaxB),maxCrossing))
```

Listing 3: pseudo code for $n \log(n)$ algorithm

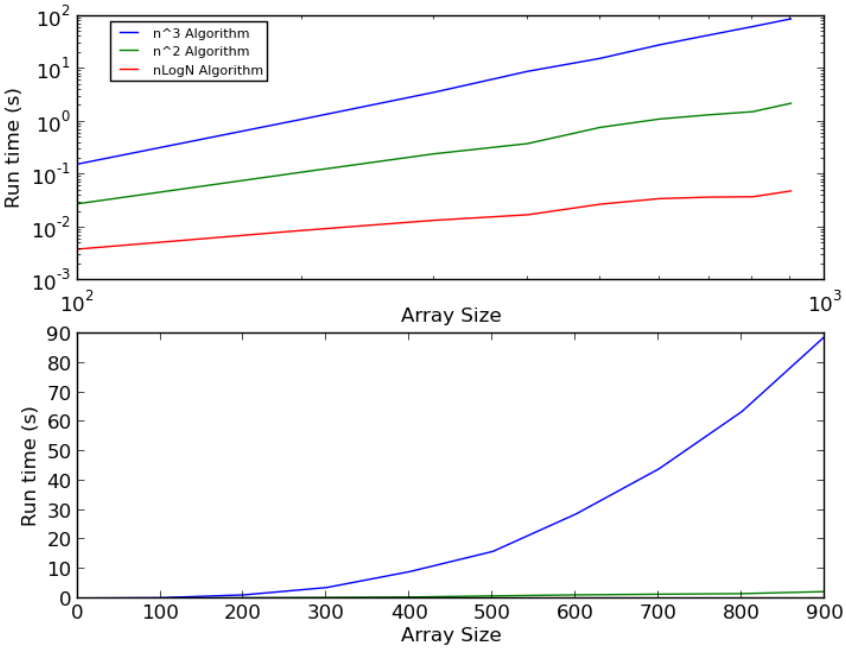

Figure 1: Plot of the three algorithms up to array size of 900, top: log/log, bottom: normal axis

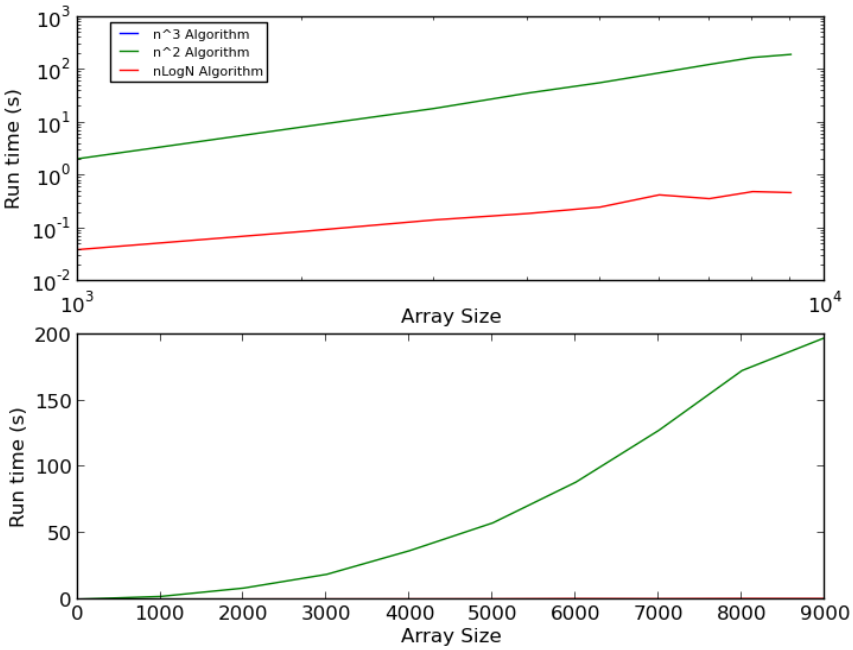

Figure 2: Plot of the three algorithms up to array size of 9000, top: log/log, bottom: normal axis