CS2035B Assignment 2: Arrays and Efficiency

Table of Contents

Identification

Robert Moir: 123456789

avg1 Source Code

```
function avg = avg1(x)

if size(x,1) == 1
    x = x.';
end
[m,n] = size(x);

for i=1:n
    avg(i) = 0;
    for j=1:m
        avg(i) = avg(i)+x(j,i);
    end
    avg(i) = avg(i)/m;
end
end
```

avg2 Source Code

```
function avg = avg2(x)

if size(x,1) == 1
    x = x.';
```

```
end
[m,n] = size(x);
avg = zeros(1,n);
i=0; j=0;
for i=1:n
    for j=1:m
        avg(i) = avg(i)+x(j,i);
    end
    avg(i) = avg(i)/m;
end
end
```

avg3 Source Code

```
function avg = avg3(x)

if size(x,1) == 1
    x = x.';
end
n = size(x,1);
avg = sum(x)/n;
end
```

stderr1 Source Code

```
function [se, xbar] = stderr1(x)

if size(x,1) == 1
    x = x.';
end
[m,n] = size(x);
xbar = avg1(x);
for i=1:n
    se(i) = 0;
    for j=1:m
        se(i) = se(i) + (x(j,i)-xbar(i))^2;
    end
    se(i) = sqrt(se(i)/(m-1));
end
end
```

stderr2 Source Code

```
function [se, xbar] = stderr2(x)

if size(x,1) == 1
```

```
    x = x.';
end
[m,n] = size(x);
xbar = mean(x);
se = zeros(1,n);
i=0;
j=0;
for i=1:n
    se(i) = 0;
    for j=1:m
        se(i) = se(i) + (x(j,i)-xbar(i))^2;
    end
    se(i) = sqrt(se(i)/(m-1));
end
end
```

stderr3 Source Code

```
function [se, xbar] = stderr3(x)

if size(x,1) == 1
    x = x.';
end
n = size(x,1);
xbar = mean(x);
se = sqrt(sum((x-xbar).^2)/(n-1));
```

Program Validation

```
%% Average computation tests
X = rand(1000,6);
y = rand(1,1000);
disp('avg1:')
avg1(X)
avg1(y)
disp('avg2:')
avg2(X)
avg2(y)
disp('avg3:')
avg3(X)
avg3(y)
disp('mean:')
mean(X)
mean(y)

%% Standard error computation tests
disp('stderr1:')
stderr1(X)
stderr1(y)
```

```
disp('stderr3:')
stderr2(X)
stderr2(y)
disp('stderr3:')
stderr3(X)
stderr3(y)
disp('std:')
std(X)
std(y)
```

```
tests

avg1:

ans =

    0.4920    0.5153    0.5123    0.4907    0.4919    0.4928


ans =

    0.4835

avg2:

ans =

    0.4920    0.5153    0.5123    0.4907    0.4919    0.4928


ans =

    0.4835

avg3:

ans =

    0.4920    0.5153    0.5123    0.4907    0.4919    0.4928


ans =

    0.4835

mean:

ans =

    0.4920    0.5153    0.5123    0.4907    0.4919    0.4928


ans =
```

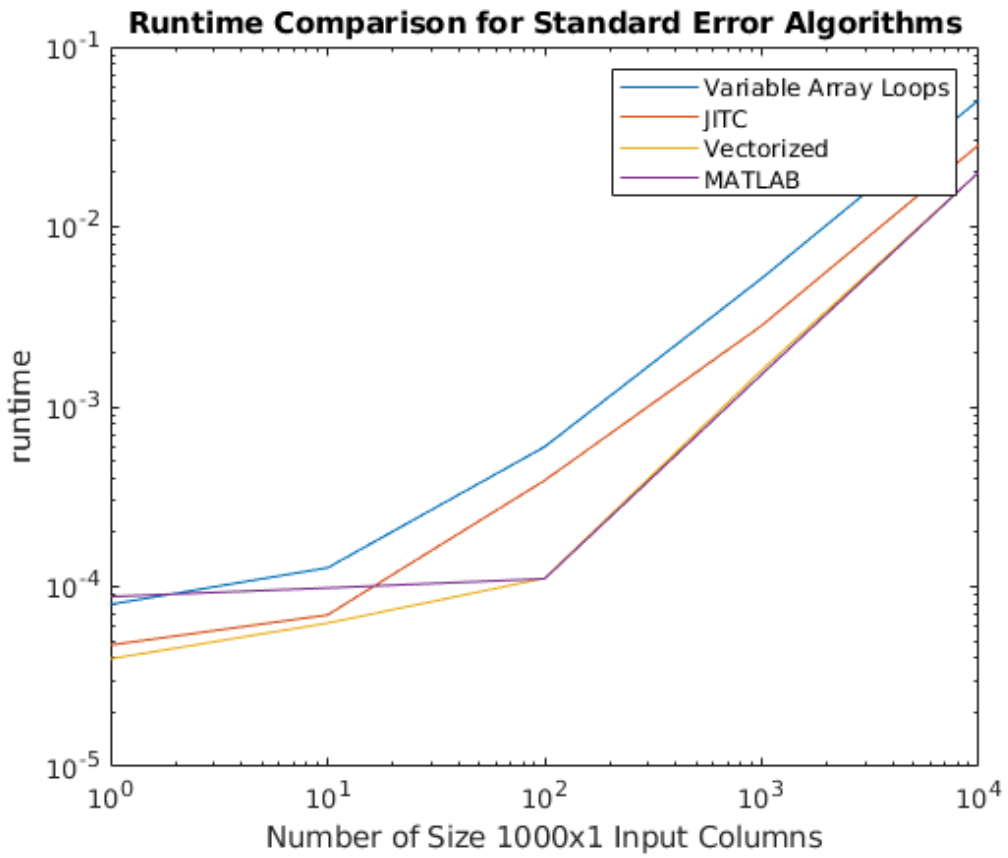
```
    0.4835

stderr1:

ans =

    0.2836    0.2860    0.2873    0.2872    0.2858    0.2910


ans =

    0.2805

stderr3:

ans =

    0.2836    0.2860    0.2873    0.2872    0.2858    0.2910


ans =

    0.2805

stderr3:

ans =

    0.2836    0.2860    0.2873    0.2872    0.2858    0.2910


ans =

    0.2805

std:

ans =

    0.2836    0.2860    0.2873    0.2872    0.2858    0.2910


ans =

    0.2805
```

Measuring Runtime Performance

```
m = 1000;
pow = 4;
```

```
points = pow+1;
n = logspace(0,pow,points);
runs = 40;
T = zeros(4,points);

for k=1:length(n)
    x = rand(m,n(k));
    for i=1:runs
        tic
        stderr1(x);
        elapsed(i) = toc;
    end
    T(1,k) = mean(elapsed);
    fprintf('stderr1 on %dx%d array: %f s\n',m,n(k),mean(elapsed));
    for i=1:runs
        tic
        stderr2(x);
        elapsed(i) = toc;
    end
    T(2,k) = mean(elapsed);
    fprintf('stderr2 on %dx%d array: %f s\n',m,n(k),mean(elapsed));
    for i=1:runs
        tic
        stderr3(x);
        elapsed(i) = toc;
    end
    T(3,k) = mean(elapsed);
    fprintf('stderr3 on %dx%d array: %f s\n',m,n(k),mean(elapsed));
    for i=1:runs
        tic
        std(x);
        elapsed(i) = toc;
    end
    T(4,k) = mean(elapsed);
    fprintf('std     on %dx%d array: %f s\n',m,n(k),mean(elapsed));
end

loglog(n,T)
legend('Variable Array Loops','JITC','Vectorized','MATLAB')
title('Runtime Comparison for Standard Error Algorithms')
xlabel('Number of Size 1000x1 Input Columns')
ylabel('runtime')
print 'stdevTiming.png' -dpng


timing

stderr1 on 1000x1 array: 0.000080 s
stderr2 on 1000x1 array: 0.000047 s
stderr3 on 1000x1 array: 0.000040 s
std     on 1000x1 array: 0.000088 s
stderr1 on 1000x10 array: 0.000127 s
stderr2 on 1000x10 array: 0.000070 s
stderr3 on 1000x10 array: 0.000063 s
```

```
std     on 1000x10 array: 0.000098 s
stderr1 on 1000x100 array: 0.000599 s
stderr2 on 1000x100 array: 0.000390 s
stderr3 on 1000x100 array: 0.000111 s
std     on 1000x100 array: 0.000111 s
stderr1 on 1000x1000 array: 0.005175 s
stderr2 on 1000x1000 array: 0.002826 s
stderr3 on 1000x1000 array: 0.001600 s
std     on 1000x1000 array: 0.001515 s
stderr1 on 1000x10000 array: 0.050721 s
stderr2 on 1000x10000 array: 0.028552 s
stderr3 on 1000x10000 array: 0.020085 s
std     on 1000x10000 array: 0.020136 s
```



Published with MATLAB® R2018a