Approximated PCA

Iteration 6

Rodrigo Arias

July 3, 2017

Reduction of space in Householder

The Householder algorithm needs 6 internal f.p. variables to perform the tridiagonalization.

Also, 3 arguments are reused to store the results:

- The matrix A with size $n \times n$
- The diagonal d with size n
- The offdiagonal o with size n

In total, 9 variables are used.

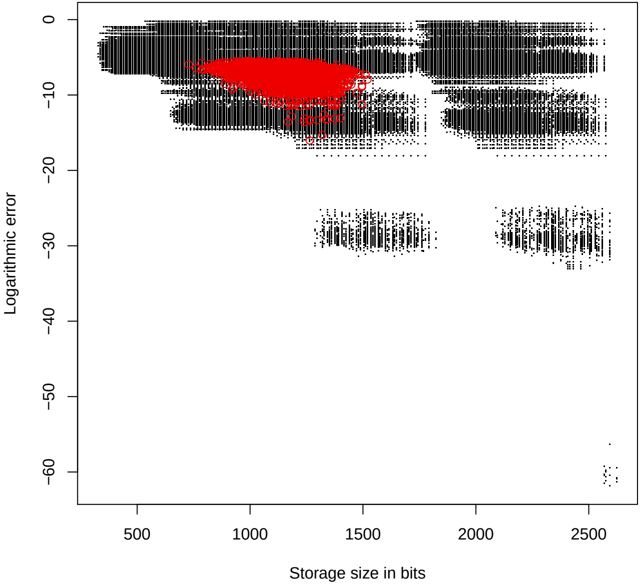
Individual precision

- The 3 arguments have the bigger impact (multiple elements) as shown by previous experiments.
- Different precisions can be set to individual elements.
- We can compare results between different configurations.

Experiment H

A new experiment can be designed, to test individual precisions.

- Each element is assigned a random precision from $C = \{8, 16, 32, 64\}$, with equal probability
- The other variables are kept at 64 bits.
- The storage size and the error is plotted.


0
−10
−20
−30
−40
−50
−60
Logarithmic error
500
1000
1500
2000
2500
Storage size in bits

$n = 5$

Results obtained

- The results obtained have a bigger error and use more size.
- It seems that there is no advantage in using different precisions in individual elements.
- The utility function can be used to compare the results

Utility function

Assuming that we are interested only in the reduction of **storage size** s, while maintaining a low error, the utility function u can be defined as:

$$u = \text{error} + \text{size}$$

However, both quantitites need to be scaled acordingly, so that a unit reduction of error, is equally good as a unit of reduction of size.

Scaling error and size

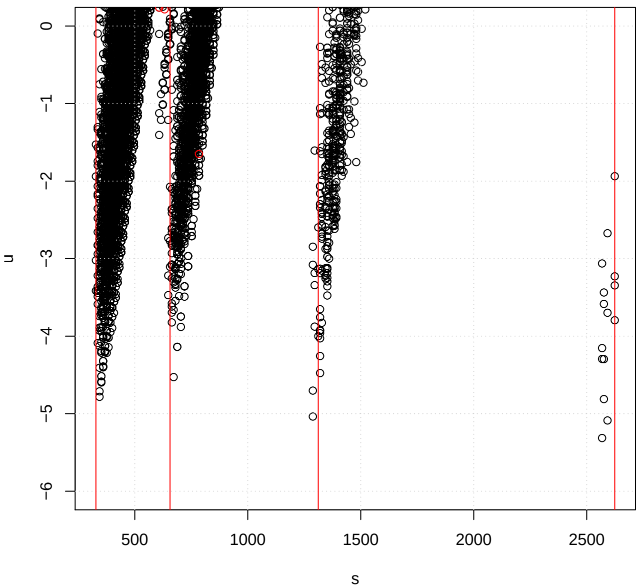
We can use the relation between error and space s that we measured experimentally.

$$\log_2 \Delta \approx -b + \alpha \log_2 n$$

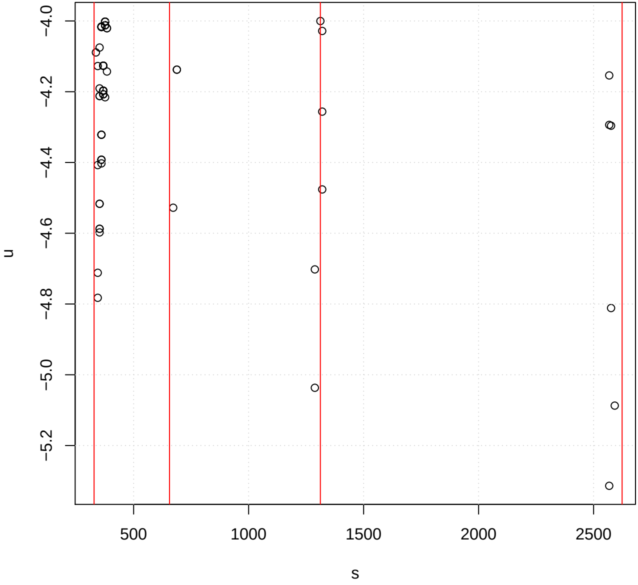
Then, u becomes

$$u(\Delta, b, n) = \log_2 \Delta + b - \alpha \log_2 n$$

where b is the mean bit-width, n is the size of the input $n \times n$ matrix, and $\alpha = 2.78857$.



$n = 5$



$n = 5$

Conlusions

The best results (from the point of view of the utility function) are those that:

- Maintain the same precision in individual elements
- Use all the variables to the same value b
- Except for the scale variable, which has almost no influence in the error.

Caveats and posible solutions

The current utility function does not measure the time nor the energy. A posible solution could be done by simulation.

- A simulation can estimate the time of the floating point units with bit-width b.
- The IO operations in RAM and cache can be simulated and measured.
- The conversion between different bit-width variables needs to be also accounted.