

CME 212: Assignment 6

Raunak Borker

The profiling has been performed using the tool CProfile. The top 5 in the list have been shown here.

Profiling Python 'only' implementation

29303807 function calls (29301565 primitive calls) in 465.674 seconds

Ordered by: internal time
List reduced from 2242 to 20 due to restriction <20>

ncalls	tottime	percall	cumtime	percall	filename:lineno(function)
4304	262.876	0.061	460.022	0.107	piv.py:19(XCorr)
7055798	171.048	0.000	171.048	0.000	{method 'reduce' of 'numpy.ufunc' objects}
7019288	13.015	0.000	197.330	0.000	/usr/lib/python2.7/dist-packages/numpy/core/fromnumeric.py:1621(sum)
7019288	8.358	0.000	179.193	0.000	/usr/lib/python2.7/dist-packages/numpy/core/_methods.py:23(_sum)
7121713	5.159	0.000	5.159	0.000	{isinstance}

The elapsed time shown by the code was: *463.54 sec*

It can be seen from the profile output that the major chunk of the time in the code is spent in executing the XCorr function. Time is also spent on the reduce method which was also concluded to be mainly called from the XCorr function. This is expected as well since this function involves the most number of computations. Hence it was chosen to write this function as a C Extension.

Profiling Python with C extension implementation

1228831 function calls (1226589 primitive calls) in 8.554 seconds

Ordered by: internal time
List reduced from 2243 to 20 due to restriction <20>

ncalls	tottime	percall	cumtime	percall	filename:lineno(function)
4304	4.215	0.001	4.215	0.001	{_pivcext.XCorr}
4	0.441	0.110	4.918	1.229	piv.py:152(Pass)
79590	0.432	0.000	0.432	0.000	{method 'reduce' of 'numpy.ufunc' objects}
4	0.364	0.091	2.181	0.545	piv.py:324(LocalFilter)
8616	0.221	0.000	1.068	0.000	/usr/lib/python2.7/dist-packages/numpy/lib/nanfunctions.py:613(nanvar)

The elapsed time shown by the code was: *7.42 sec*

As it can be seen from these results that XCorr was indeed the bottleneck. Although it's still the bottleneck the overall performance is ~62x faster.

All the results were obtained on corn12 machine.