

# Oscar Teeninga

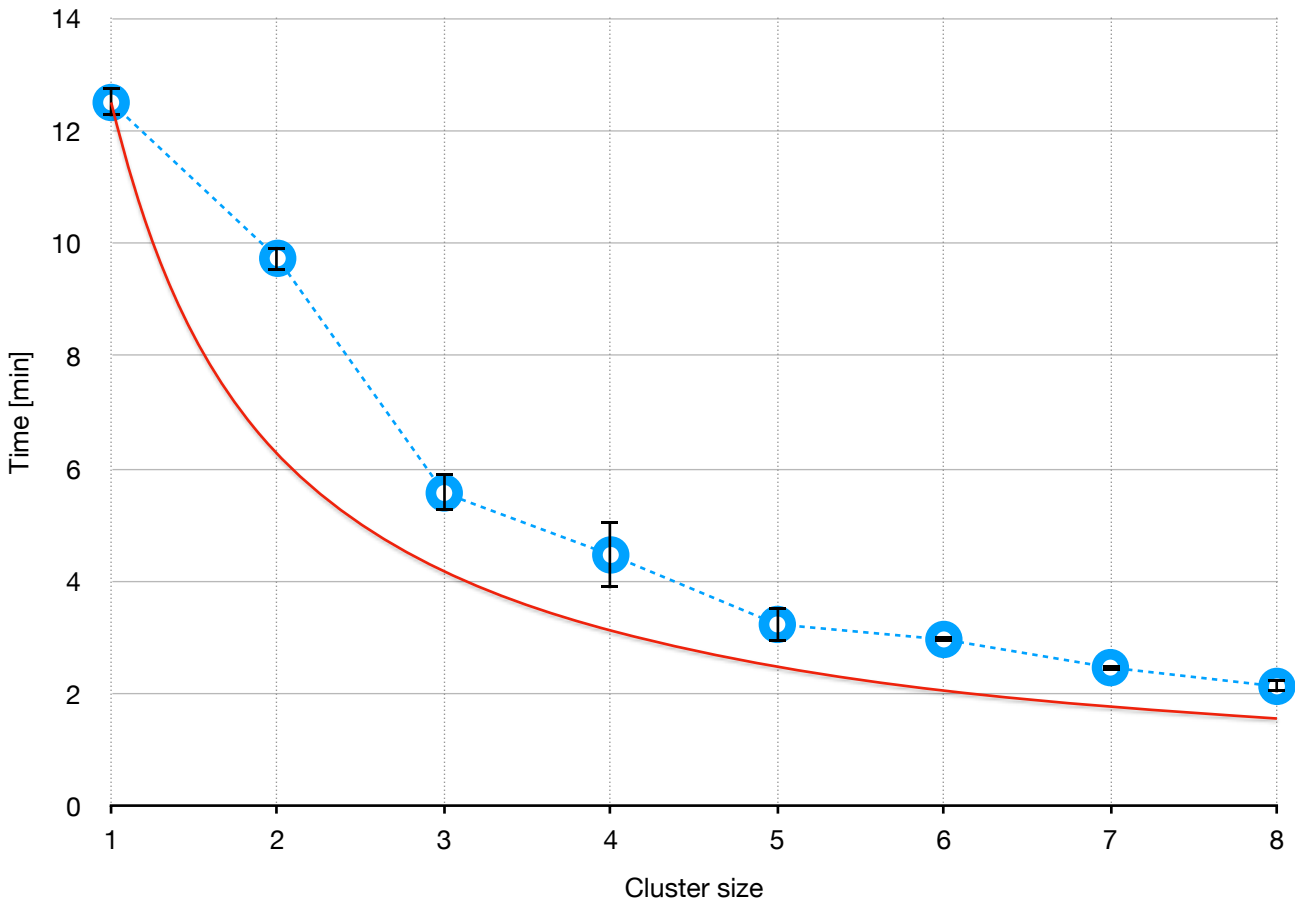
MapReduce  
m4.large

## Wnioski

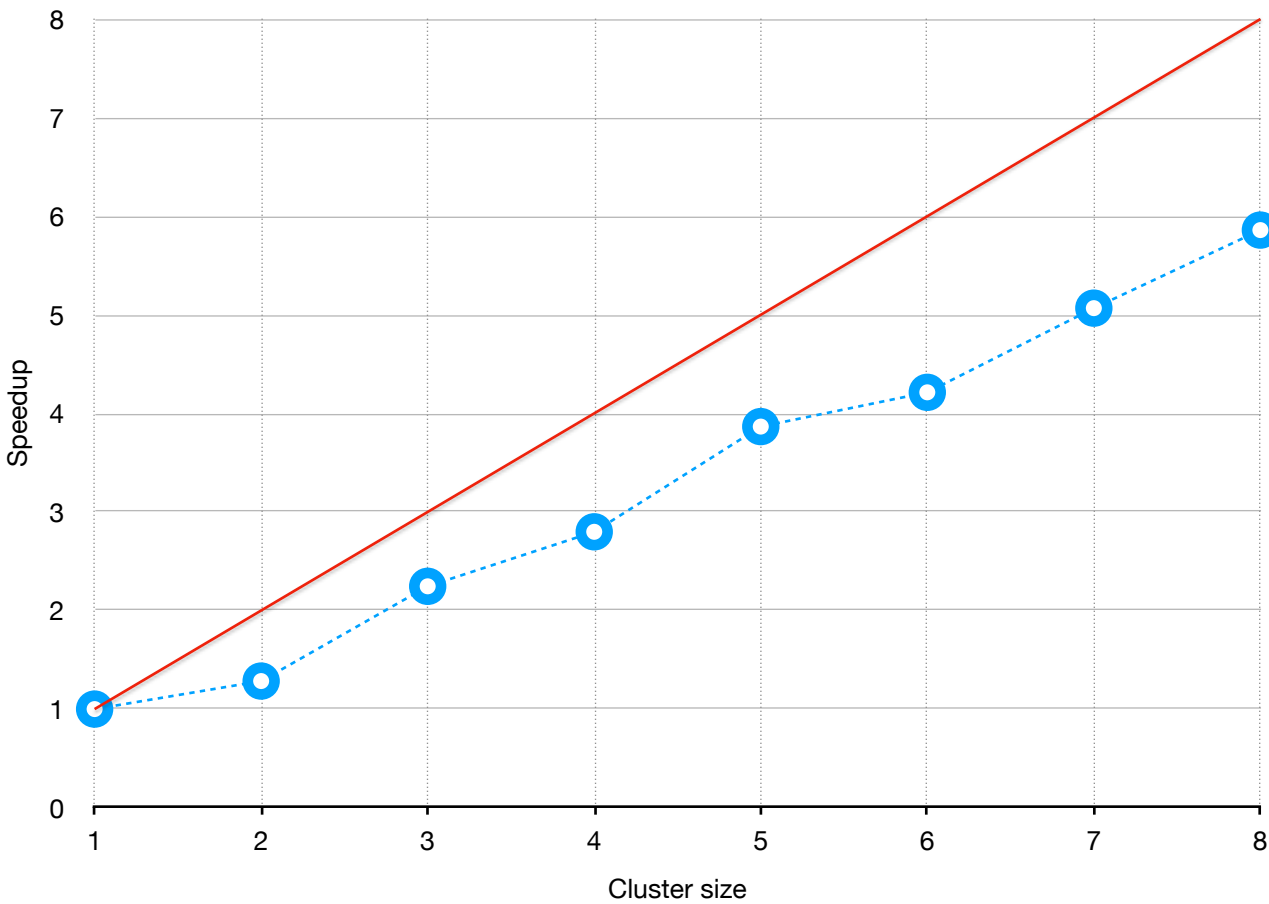
Zwiększenie rozmiaru klastra zgodnie z oczekiwaniami sprawiła, że obliczenia były wykonywane szybciej. Bardzo ciekawy jest nieproporcjonalnie duży koszt dla dwóch instancji. Koszt był w ogólności stały niezależnie od liczby procesorów. W przypadku m4.large każda instancja ma 2 rdzenie. Czas dla wersji sekwencyjnej: 6s

Cluster size	1	2	3	4	5	6	7	8
Core	2	4	6	8	10	12	14	16
Time [min]	12,4	9,5	5,8	3,8	3,6	3,0	2,4	2,2
	12,8	9,8	5,7	4,8	3,1	2,9	2,5	2,0
	12,3	9,9	5,2	4,8	3,0	3,0	2,5	2,2
Stdev [min]	0,26	0,21	0,32	0,58	0,32	0,06	0,06	0,12
Average [min]	12,50	9,73	5,57	4,47	3,23	2,97	2,47	2,13
Perfect [min]	12,50	6,25	4,17	3,13	2,50	2,08	1,79	1,56
Speedup	1,00	1,28	2,25	2,80	3,87	4,21	5,07	5,86
Perfect	1,00	2,00	3,00	4,00	5,00	6,00	7,00	8,00
All time [min]	12,50	19,47	16,70	17,87	16,17	17,80	17,27	17,07
Cost [min]	0,00	6,97	4,20	5,37	3,67	5,30	4,77	4,57

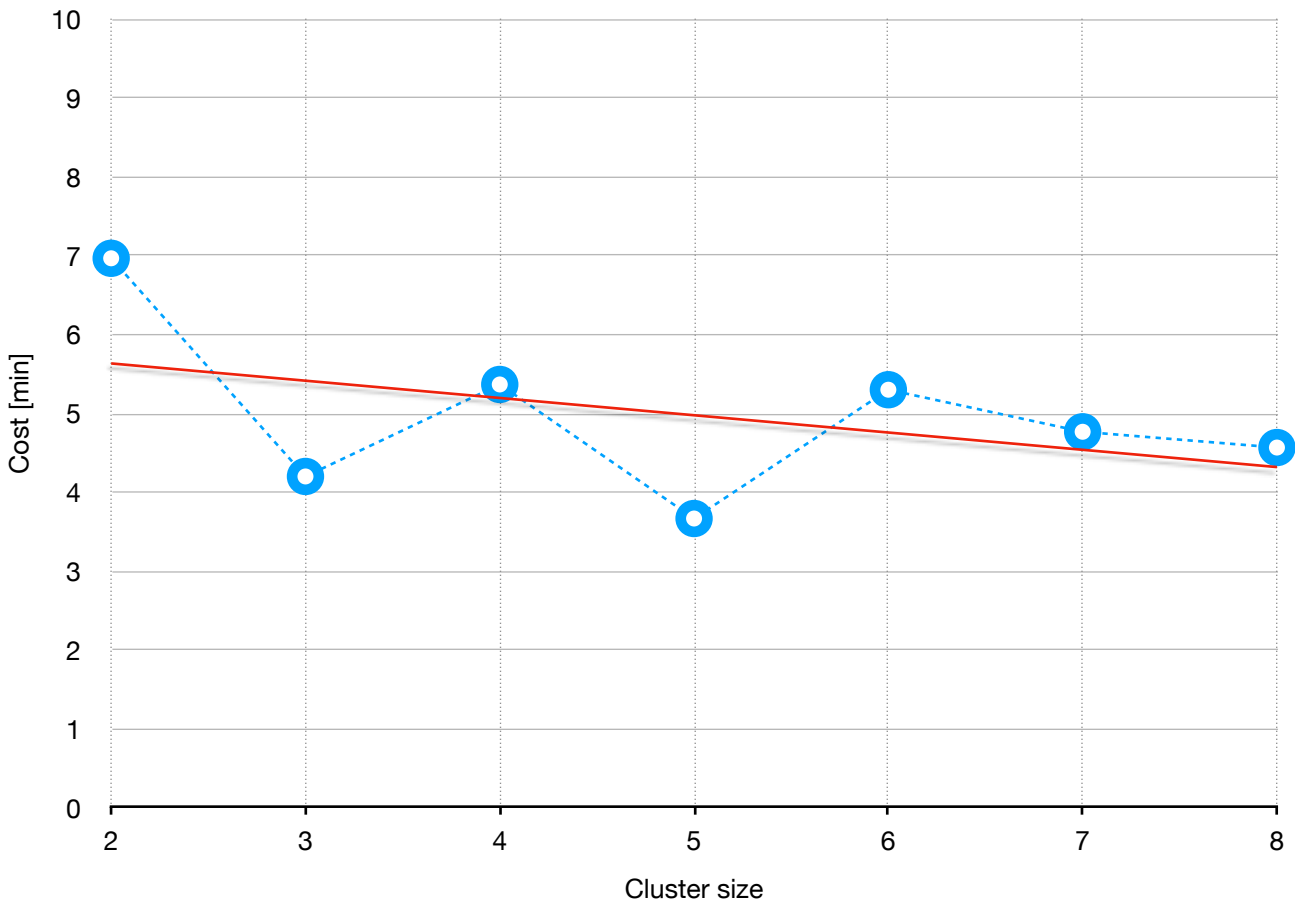
Time - m4.large - 100



Speedup - m4.large - 100



Time - m4.large - 100



# Oscar Teeninga

MapReduce  
m5.xlarge

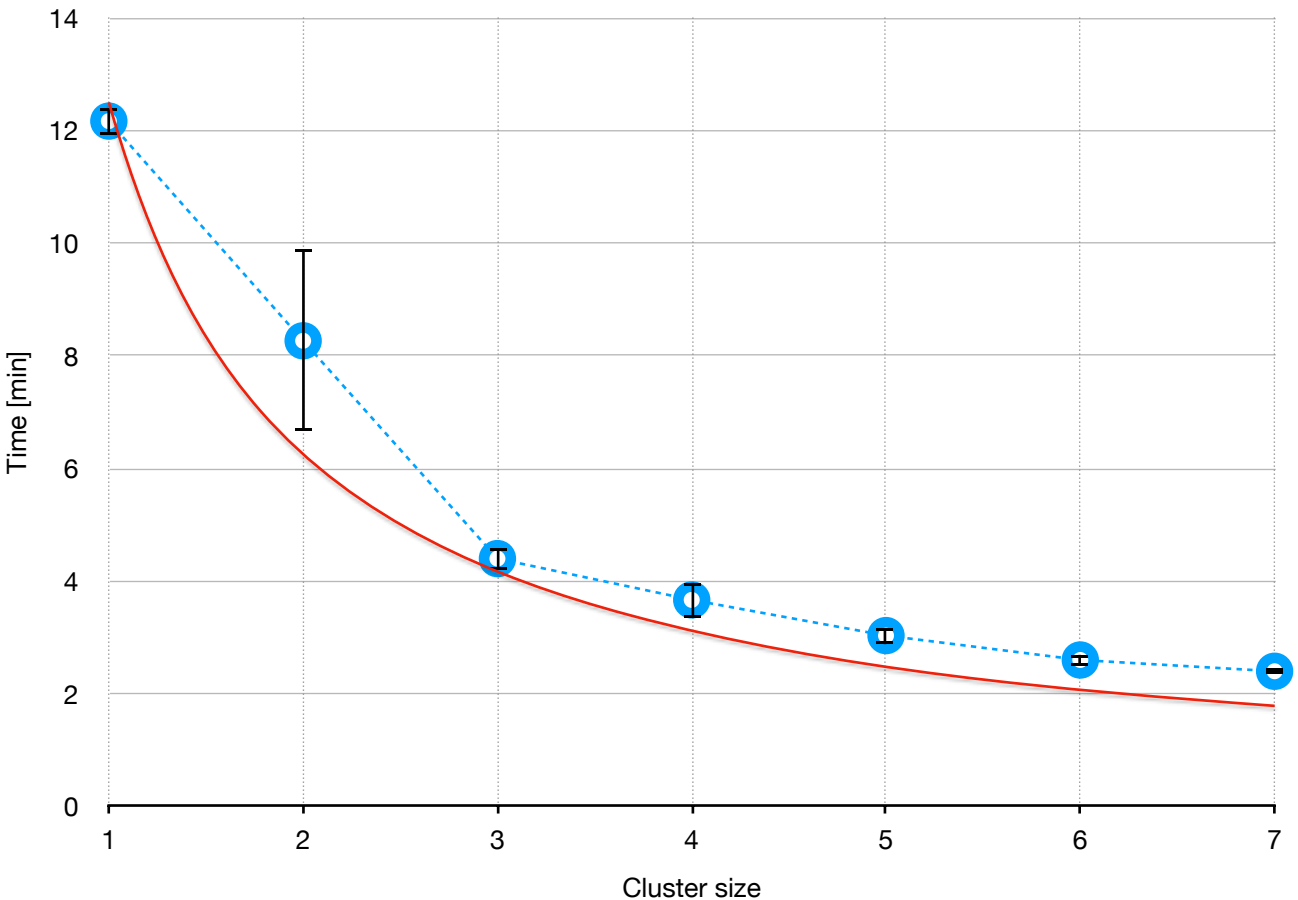
## Wnioski

Zwiększenie rozmiaru klastra zgodnie z oczekiwaniami sprawiła, że obliczenia były wykonywane szybciej. Bardzo ciekawy jest nieproporcjonalnie duży koszt dla dwóch instancji. Koszt w tym przypadku rósł wraz ze wzrostem liczby procesorów. Czas wykonania sekwencyjnego algorytmu to 8s.

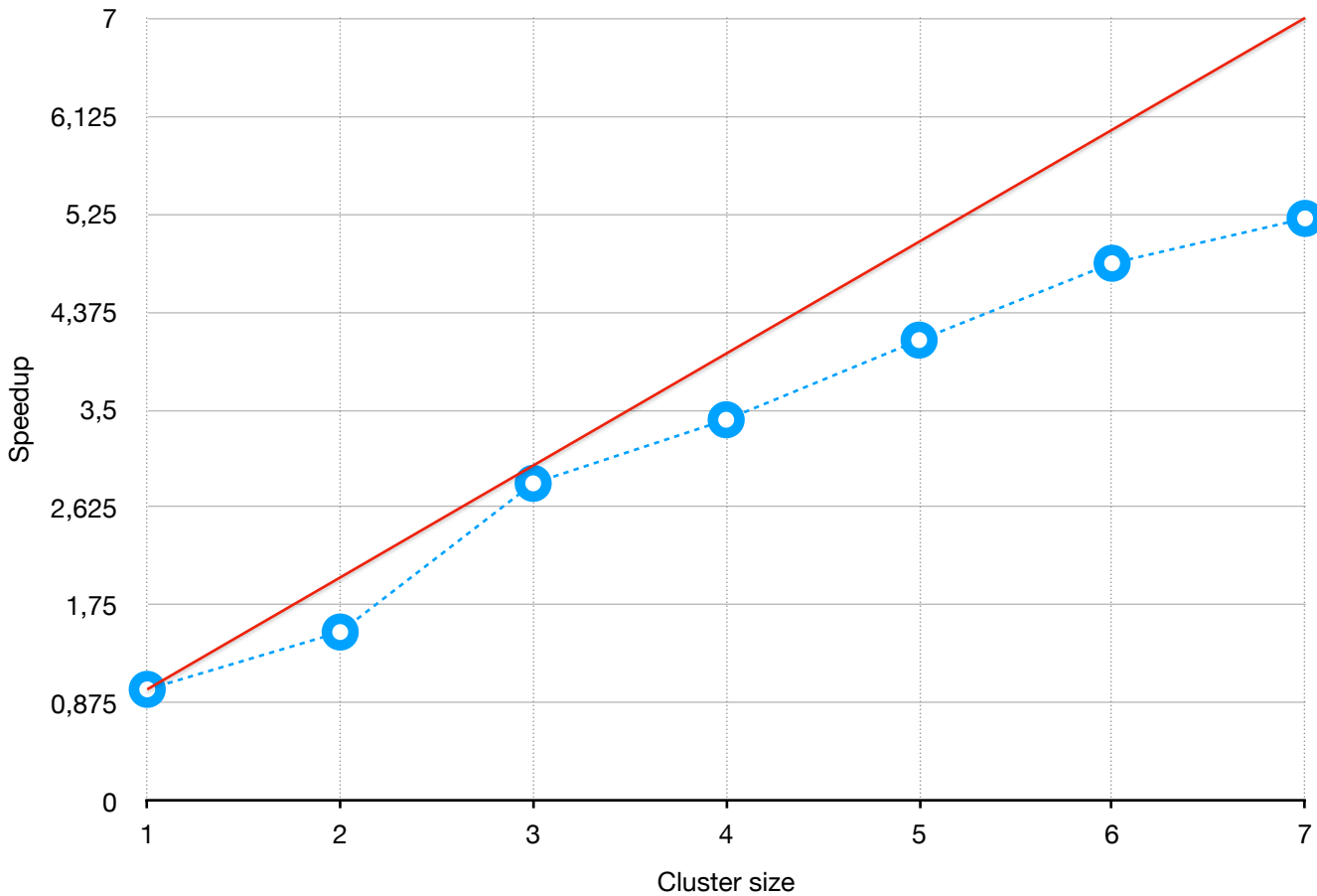
m5.xlarge n = 200

Cluster size	1	2	3	4	5	6	7
Core	4	8	12	16	20	24	28
Time [min]	12,2	6,4	4,3	4	3	2,7	2,4
	11,9	9,3	4,3	3,6	2,9	2,6	2,4
	12,4	9,1	4,6	3,4	3,2	2,5	2,4
Stdev [min]	0,25	1,62	0,17	0,31	0,15	0,10	0,00
Average [min]	12,17	8,27	4,40	3,67	3,03	2,60	2,40
Perfect [min]	12,50	6,25	4,17	3,13	2,50	2,08	1,79
Speedup	1,00	1,51	2,84	3,41	4,12	4,81	5,21
Perfect	1,00	2,00	3,00	4,00	5,00	6,00	7,00
All time [min]	12,17	16,53	13,20	14,67	15,17	15,60	16,80
Cost [min]	0,00	4,37	1,03	2,50	3,00	3,43	4,63

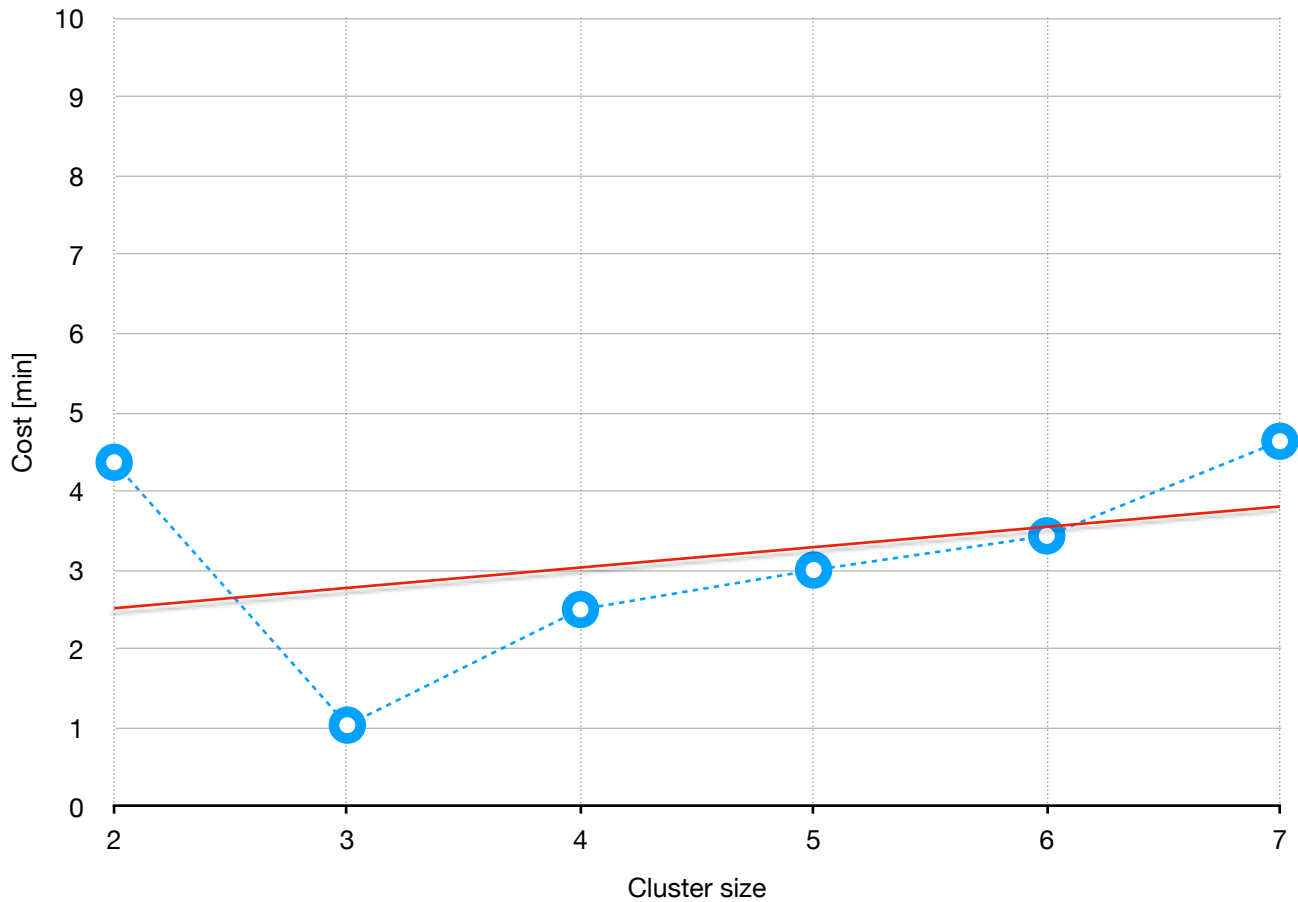
Time - m5.xlarge - 200



Speedup - m5.xlarge - 200



Cost - m5.xlarge - 200



# Oscar Teeninga

## MapReduce Podsumowanie

Punktem odniesienia nie była najlepsza wersja sekwencyjna na master, ponieważ czas ten był nawet do kilkuset krotnie lepszy niż pojedynczy klaster, co oznacza, że nawet w przypadku idealnego przyspieszenia niezbędne do osiągnięcia porównywalnego wyniku byłoby użycie kilkuset rdzeniowej konfiguracji, która w wersji studenckiej jest zablokowana (ograniczenie do 32 rdzeni).

Dopasowałem rozmiar problemu do liczby procesorów na instancję. W przypadku m4.large problem był rozmiaru 100, natomiast instancja zawierała 2 procesory - analogicznie dla m5.xlarge było to odpowiednio 200 i 4.

Ciekawe jest to, że w przypadku większego rozmiaru osiągnęliśmy bardziej oczekiwany koszt oraz generalnie mniejszy niż dla problemu mniejszego. Dla m4.large aproksymacja kosztu wskazuje, że jest to tendencja malejąca, co jest całkowicie nieoczekiwane.

Ze względu na ograniczenie liczby procesorów do 32, nie byłem w stanie określić eksperymentalnie liczby procesorów potrzebnych do określenia COST.

