

## A comparison of matching algorithms on simulated data

```

1-step <- Significant(F1)
        & Not-Signif(F12)
2-step <- Significant(F2)
        & Not(1-step)
mean <- Not(1-step or 2-step)
    
```

Algo 1

Fdr = 16%

True Step	Step Found		
	0	1	2
0	1845	67	88
1	2	1996	2
2	6	27	1967

```

2-step <- Significant(F12)
1-step <- Significant(F1)
        & Not(2-step)
mean <- Not(1-step or 2-step)
    
```

Algo 2

(backward stepwise)

Fdr = 32%

True Step	Step Found		
	0	1	2
0	1593	67	340
1	2	1996	2
2	1	27	1972

Table 1: Step height =  $5\sigma$ . Number of timepoints = 15. A total of 2000 random data, 2000 one step data and 2000 two step data with random step positions. Steps found using *StepMiner* algorithm.

The results of *StepMiner* depend on the order in which various patterns are matched. The best order is not necessarily obvious *a priori*. This appendix describes an empirical evaluation on simulated data, in which noise has been added to known patterns.

Four different algorithms are described in Table 1 and Table 2. The algorithms were compared using a simulated time course with 15 time points. The data consisted of 2,000 “genes” consisting of  $N(0, 1)$  Gaussian noise; 2,000 one-step genes and 2000 two-step genes with added Gaussian noise ( $N(0, 1)$ ) and a step height of  $5\sigma$  ( $\sigma = 1$ ). Step positions are random.

The FDR was measured on a 10,000 artificial genes and 15 time points with 4,000 one-step genes (2,000 up-regulated, 2,000 down-regulated), 4,000 two-step genes (2,000 “up then down” and 2,000 “down then up”) and 2,000 constant genes.

The False Discovery Rate (FDR) at a p-value threshold of 0.05 was estimated by computing the average number of one-step or two-step patterns discovered in 100 random permutations of the columns divided by the actual number of one-steps or two-steps in the data. Note that this FDR doesn’t capture the mis-classification of one-step to two-step or *vice versa*.

Table 1 and Table 2 describe the steps that were found using *StepMiner* on this data. Algorithm 2 finds more false two steps (340 two steps) in the random Gaussian data than algorithm 1 (88 two steps). This explains some of the increase in FDR to 32% from 16%. Otherwise, Algorithms 1 and 2 performed similarly.

Algorithm 3 mis-classifies many two-step genes as one-step genes. It has slightly fewer mis-classifications of noise as one-step or two-step, and has a slightly better FDR than algorithm 1.

The standard forward stepwise algorithm as described in Table 2 misses very many two step patterns compared to algorithm

```

1-step <- Significant(F1)
2-step <- Significant(F12)
        & Significant(F2)
        & Not(1-step)
mean <- Not(1-step or 2-step)

```

Algo 3

Fdr = 15%

True Step	Step Found		
	0	1	2
0	1856	72	72
1	2	1997	1
2	6	609	1385

```

mean <- Not-Significant(F1)
1-step <- Not-Significant(F12)
2-step <- Not(1-step or mean)

```

Algo 4

(Forward stepwise algorithm)

Fdr = 7%

True Step	Step Found		
	0	1	2
0	1928	67	5
1	3	1996	1
2	1391	27	582

Table 2: Step height =  $5\sigma$ . Number of timepoints = 15. A total of 2,000 random data, 2,000 one-step data and 2,000 two-step data with random step positions. Steps found using standard forward stepwise algorithm.

1.

It is also illuminating to look at the behavior of the algorithms as the number of time points increases. The number of correct one-step genes and two-step genes versus the the number of time points are plotted. Figure 1, 2, 3 and 4 shows the number of correct one-step and two-step matches as the number of time points increases from 7 to 15. As can be seen in Figure 1, 2 and 4 the number of correct classifications monotonically increases with the number of time points. However, algorithm 3 misses *more* two-step genes as the number of time points increases.

Algorithm 1 has fewer two-steps from the random Gaussian data than algorithm 2. Further, Algorithm 1 has fewer misclassifications from two step to one step than algorithm 3. The performance of algorithm 1 increases as the number of time points increase. Based on these observations, we conclude that algorithm 1 achieves the desirable balance between the number of true discoveries false discoveries.

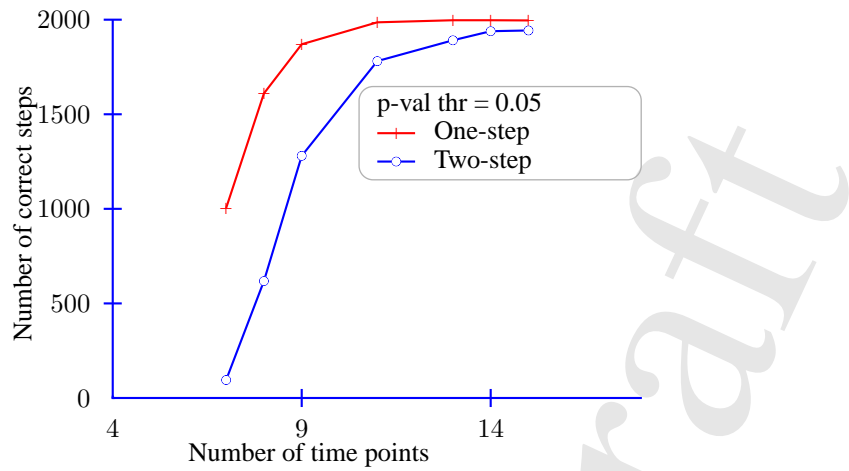


Figure 1: Algorithm 1 : Sensitivity of StepMiner to the number of time points, using random step positions and step size  $5\sigma$ . A total of 2000 one-step and 2000 two-step functions were used in the analysis.

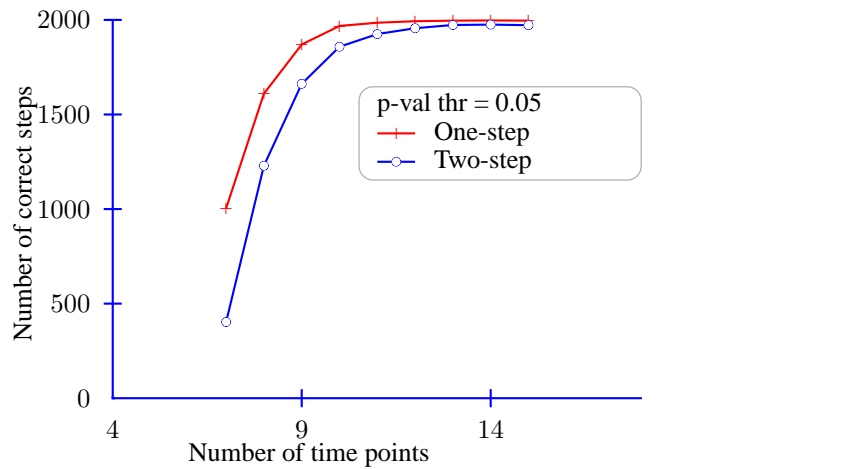


Figure 2: Algorithm 2 : Sensitivity of StepMiner to the number of time points, using random step positions and step size  $5\sigma$ . A total of 2000 one-step and 2000 two-step functions were used in the analysis.

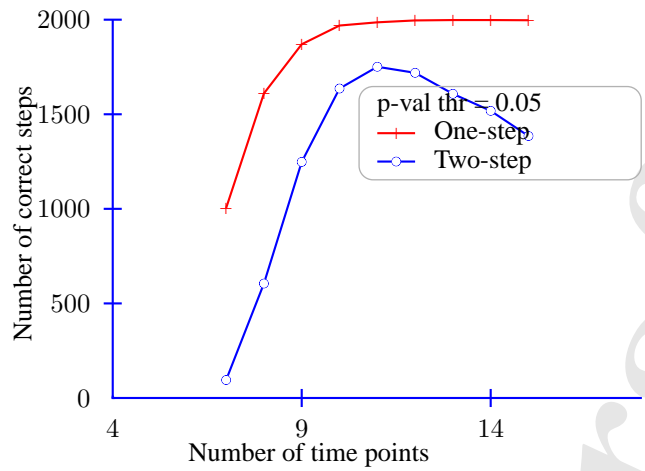


Figure 3: Algorithm 3: Sensitivity of StepMiner to the number of time points, using random step positions and step size  $5\sigma$ . A total of 2000 one-step and 2000 two-step functions were used in the analysis.

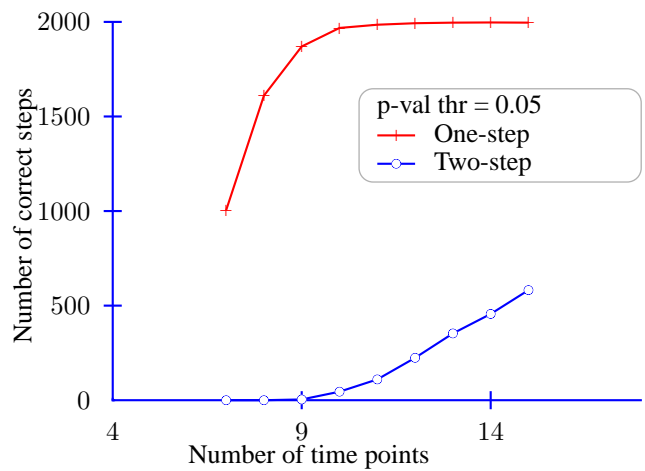


Figure 4: Algorithm 4: Sensitivity of StepMiner to the number of time points, using random step positions and step size  $5\sigma$ . A total of 2000 one-step and 2000 two-step functions were used in the analysis.