Algunas Variantes del Algoritmo PC

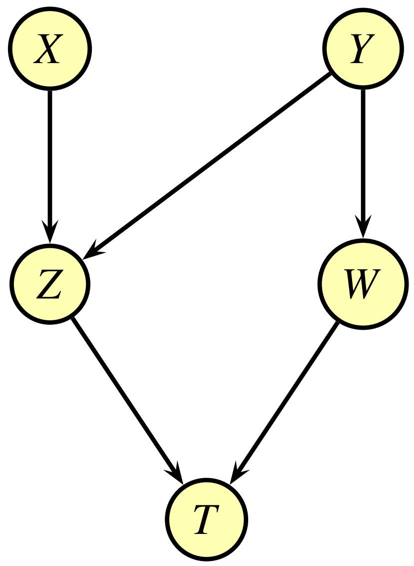
Manuel Gómez-Olmedo y Serafín Moral
Granada

The Algorithm Structure

1. Find a graph pattern (gp): an undirected graph
2. Find some head to head links by separators
3. Orient the rest of links without producing cycles

Remark: There is some degree of arbitrariness and sometimes, though independences can be represented by a DAG the direction of the arrows is counterintuitive with causality.

Graph Pattern: The Basic Condition



Two nodes, X and Y, are connected if and only if there is no subset S_{XY} of the set of vertices V such that $I(X,Y|S_{X,Y})$.

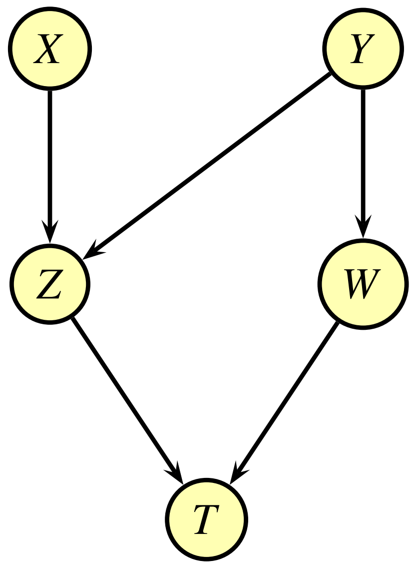
We could try to discover the graph pattern following this criterion, but it will be inefficient (too many tests) and inaccurate (conditioning to many variables).

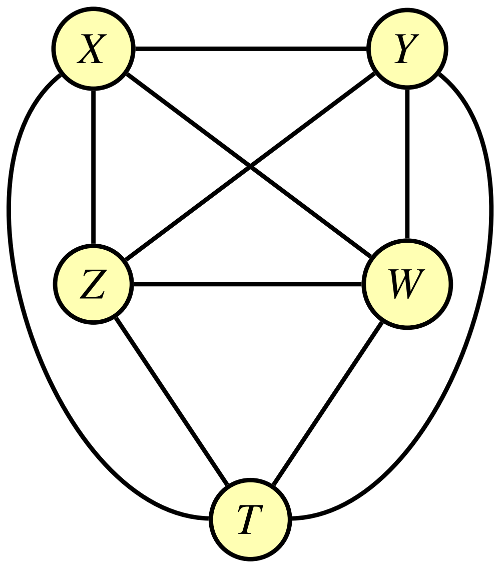
Finding the Graph Pattern

V is the set of nodes, and every independence relationships can be tested. Each node has a set of adjacent nodes ADJ_X.

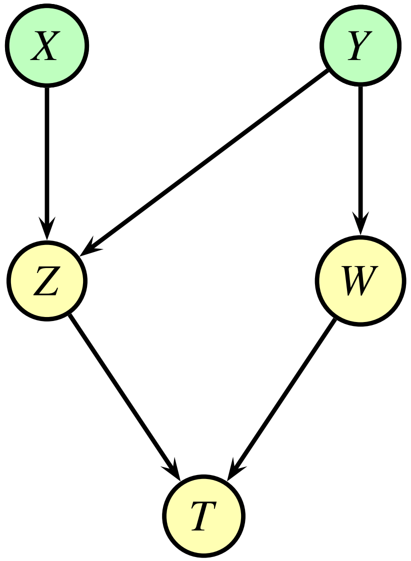
```
1.  Start with a complete undirected graph gp
2.  i = 0
3.  Repeat
  4.  For each X ∈ V
    5.  For each Y ∈ ADJ_X
      6.  Determine if there is S ⊆ ADJ_X − {Y} with |S| = i
          and I(X,Y|S)
      7.  If this set exists
        8.  Make S_XY = S
        9.  Remove X − Y link from gp
  10.  i = i + 1
11.  Until |ADJ_X| ≤ i,   ∀X
```

Example



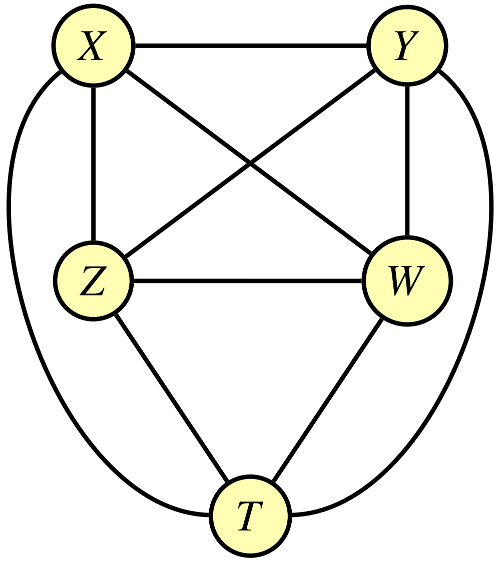


Example

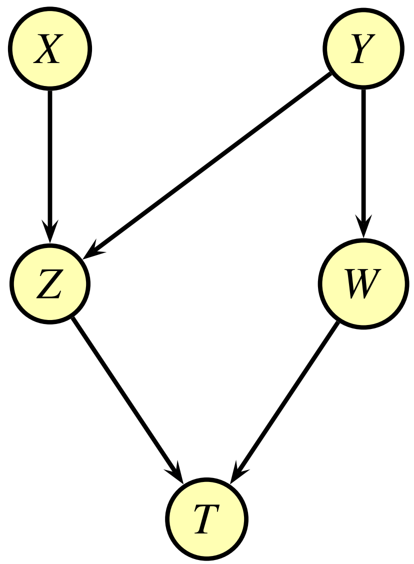


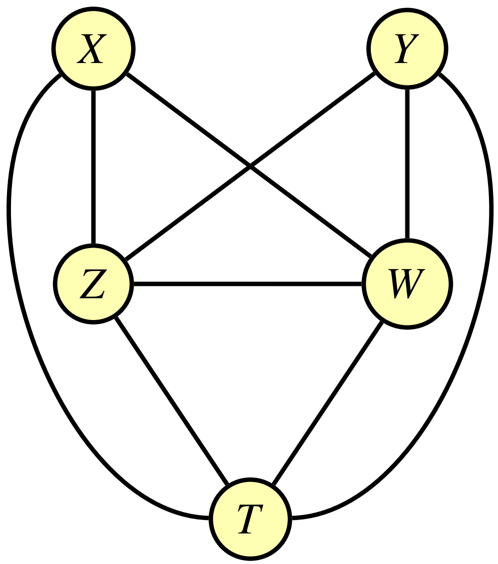
YES

$i = 0$



Example



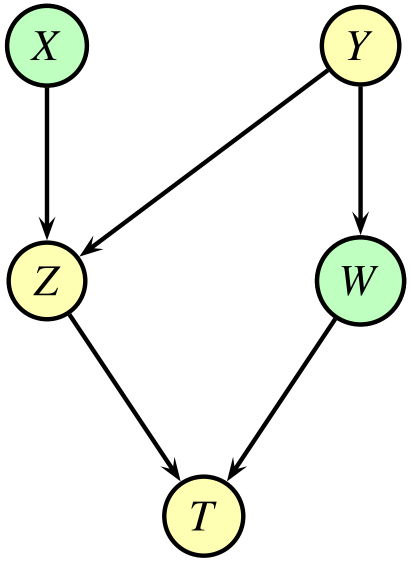


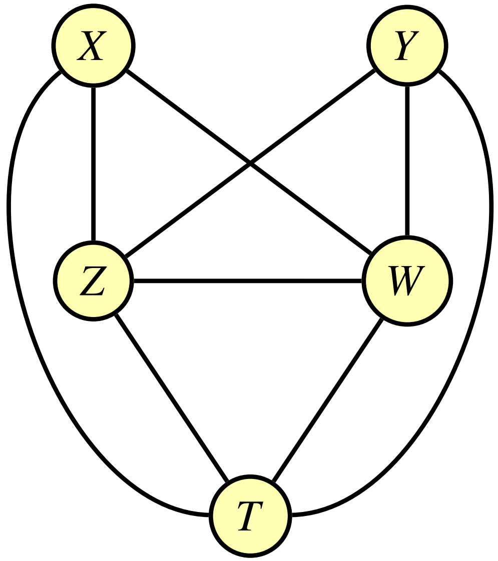
$i = 0$

$S_{X,Y} = \emptyset$

Example

YES

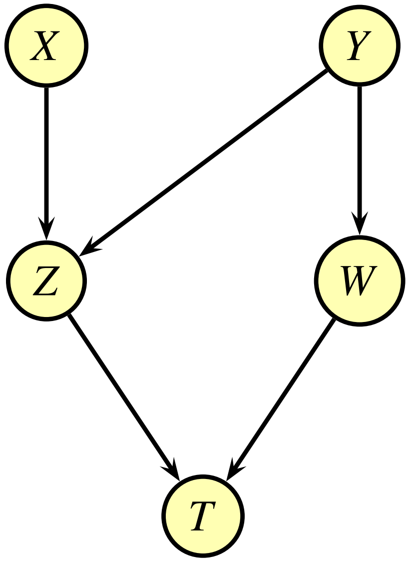


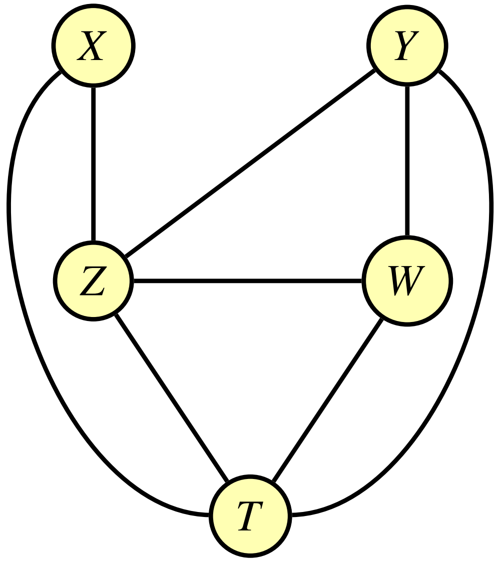


$i = 0$

$S_{X,Y} = \emptyset$

Example



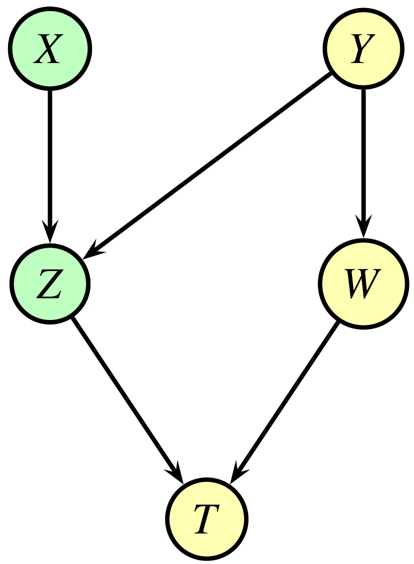


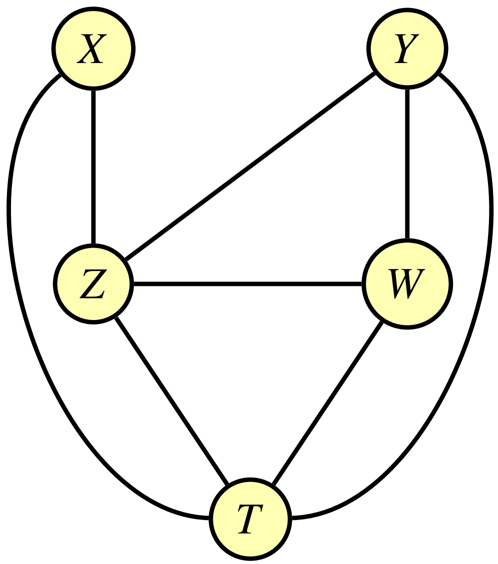
$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example

NO

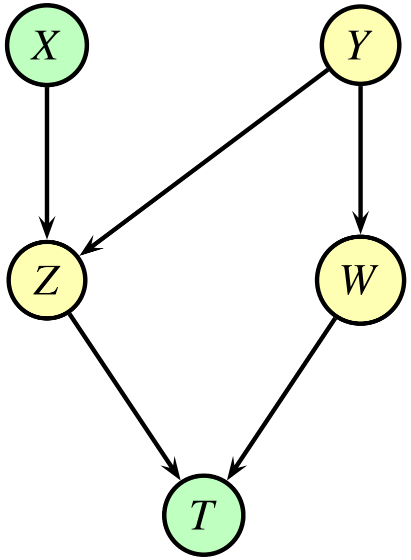


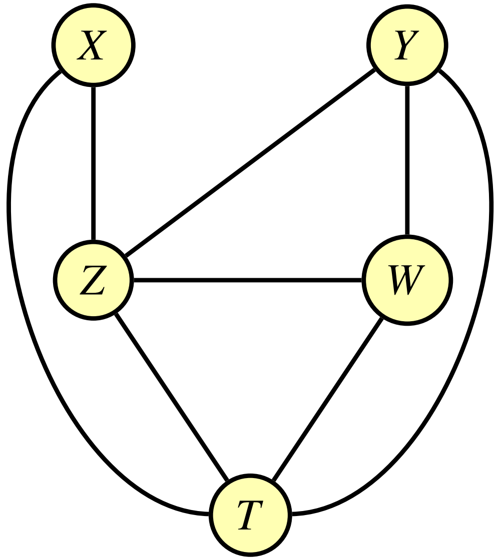


$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



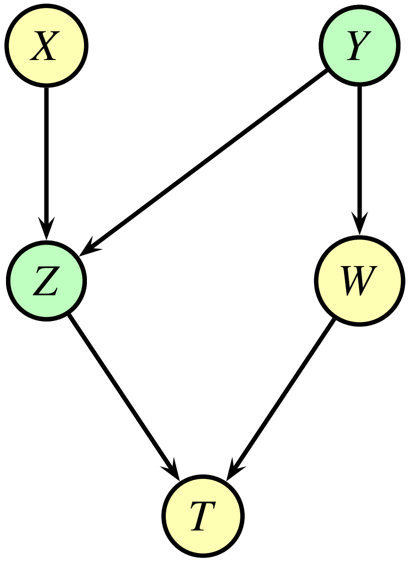


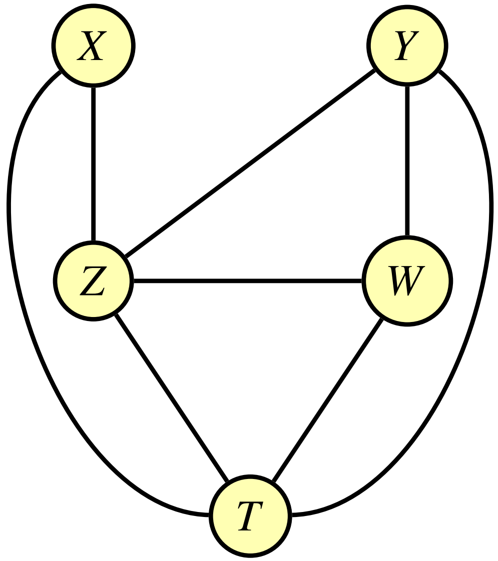
NO

$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



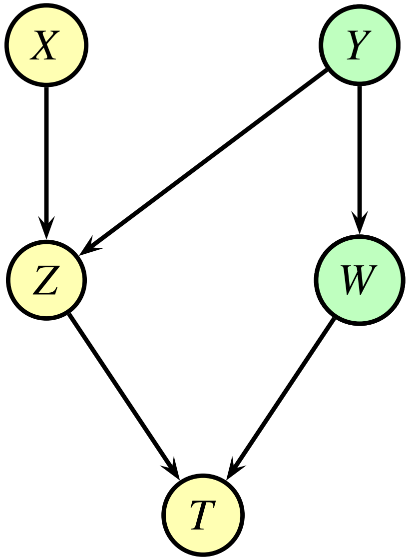


NO

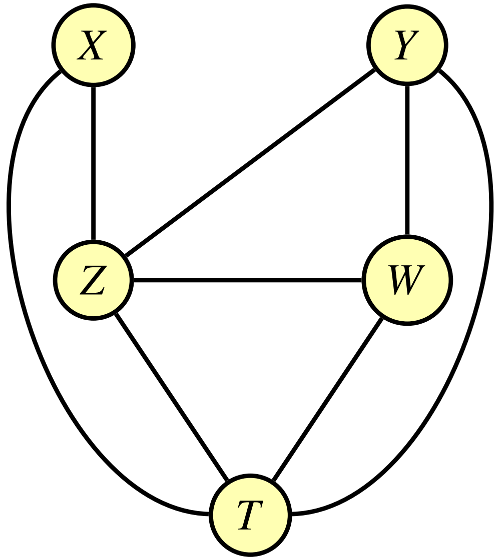
$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



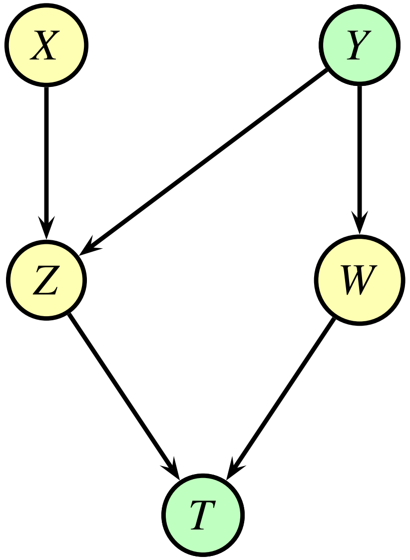
NO



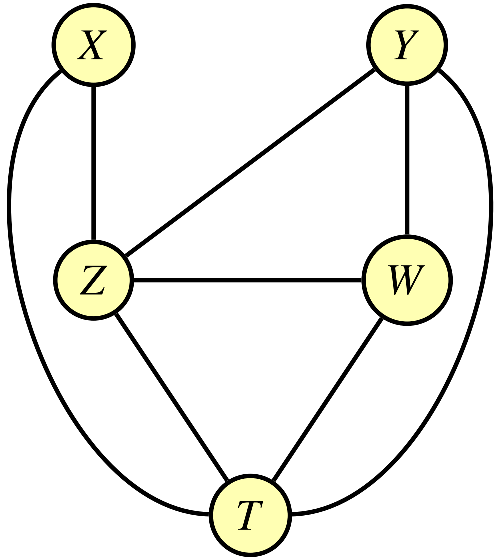
$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



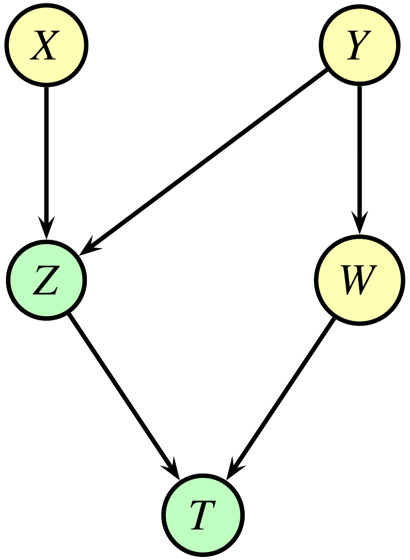
NO

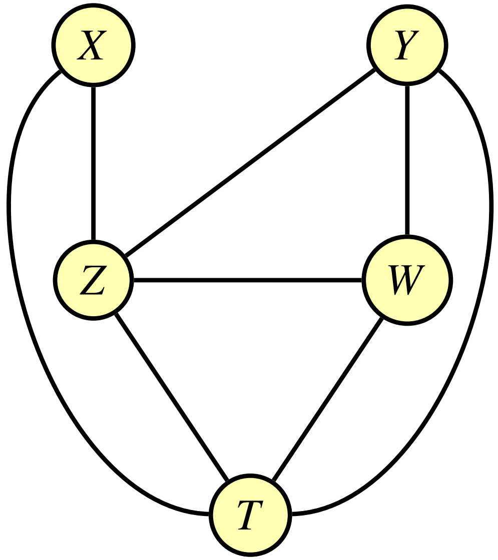


$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



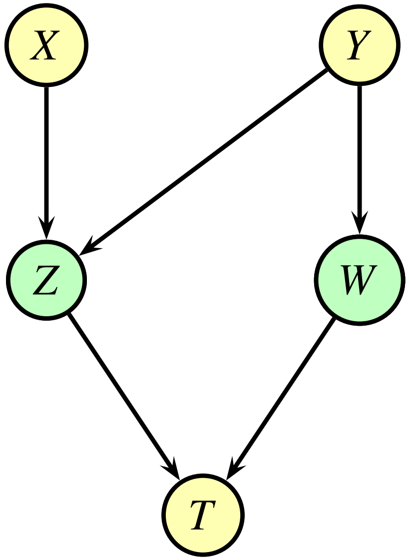


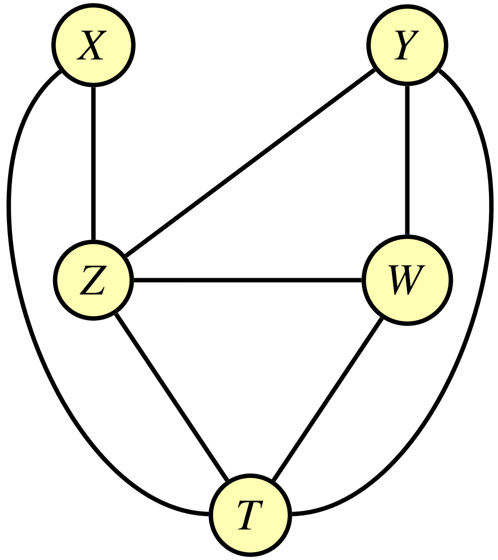
NO

$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



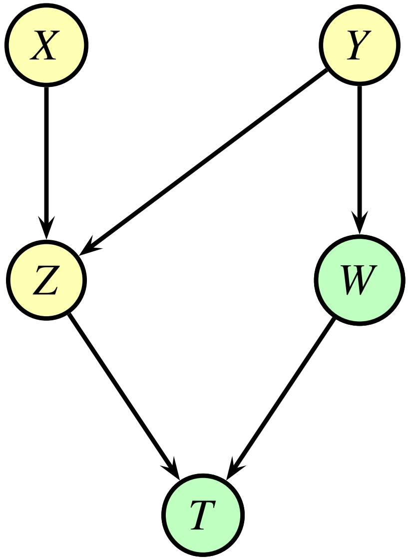


NO

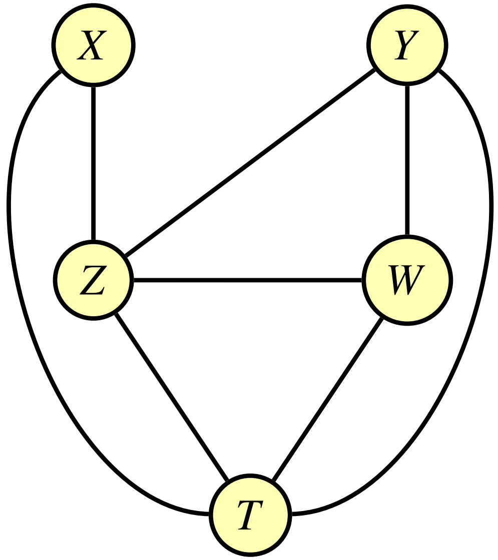
$i = 0$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



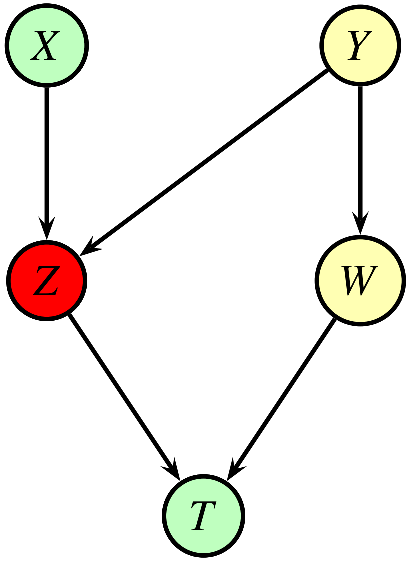
NO



$i = 0$

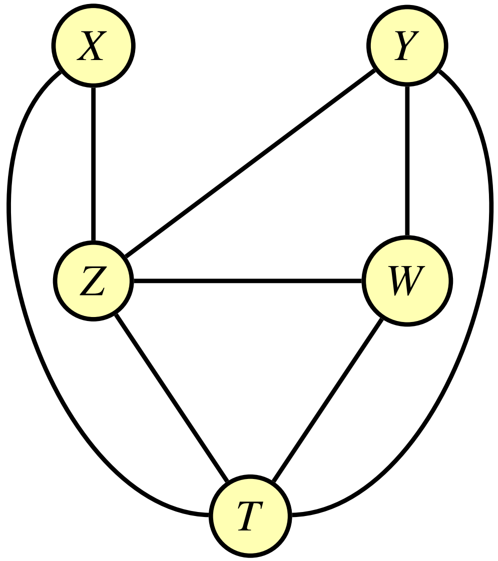
$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



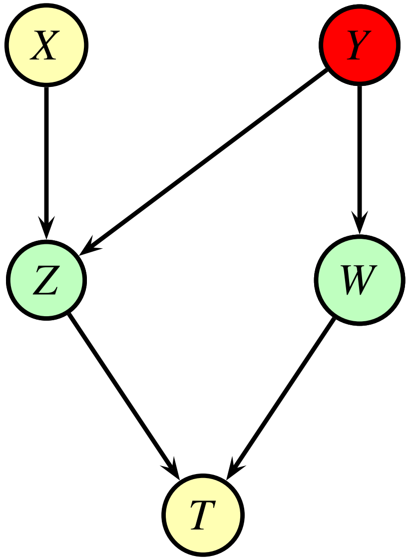
NO

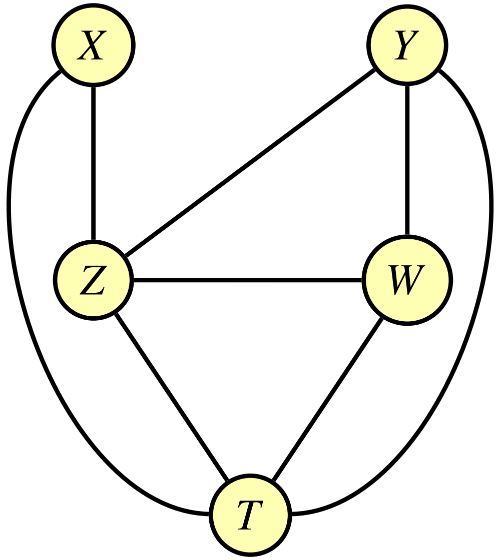
$i = 1$



$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



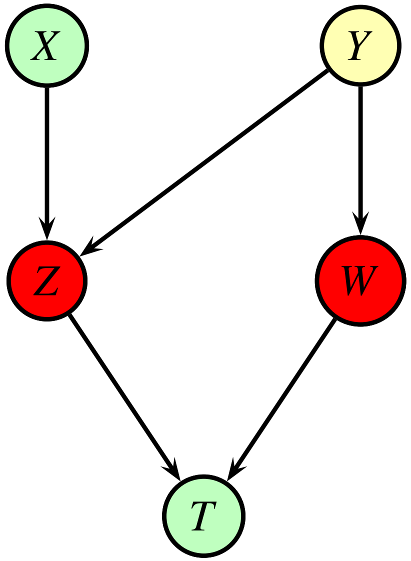


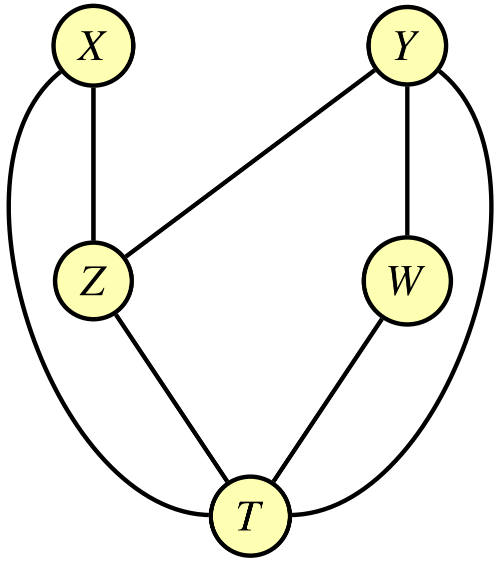
YES

$i = 1$

$S_{X,Y} = \emptyset$
$S_{X,W} = \emptyset$

Example



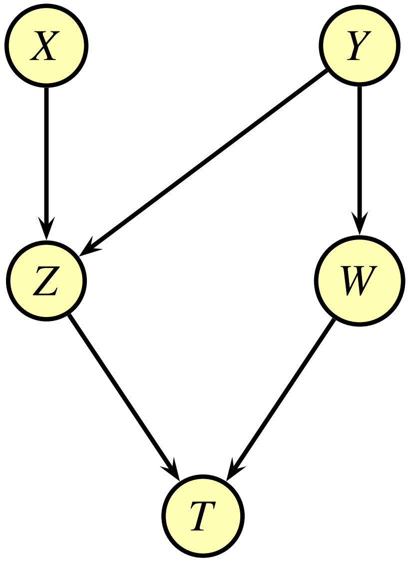


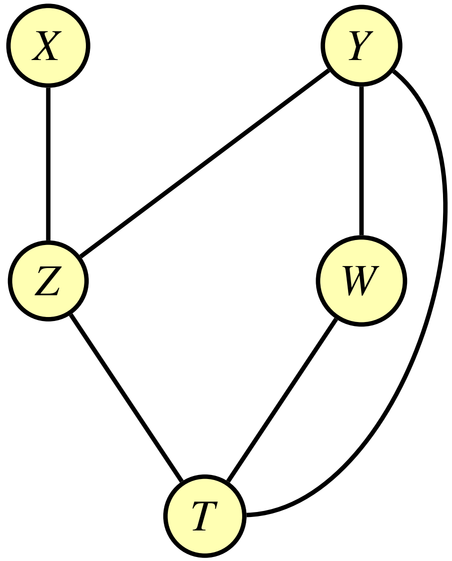
YES

$i = 2$

$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$

Example



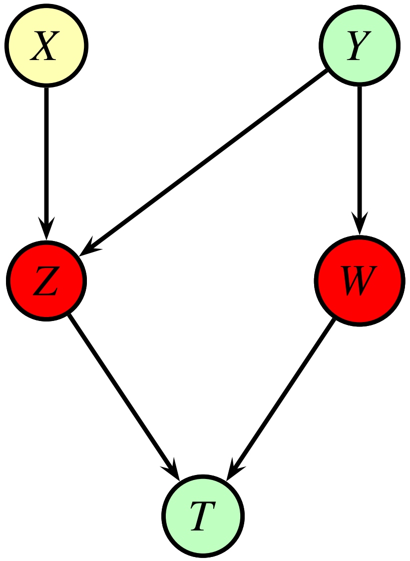


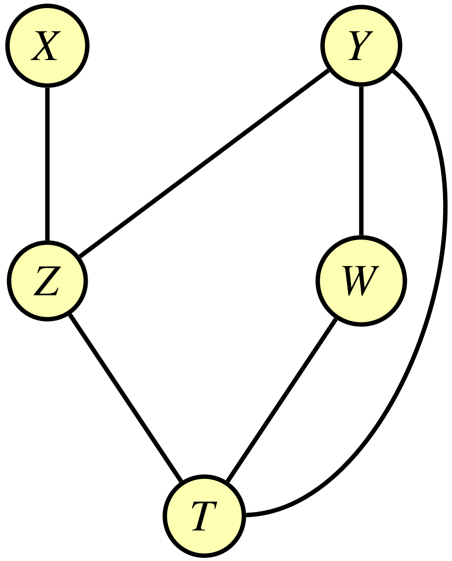
YES

$i = 2$

$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

Example

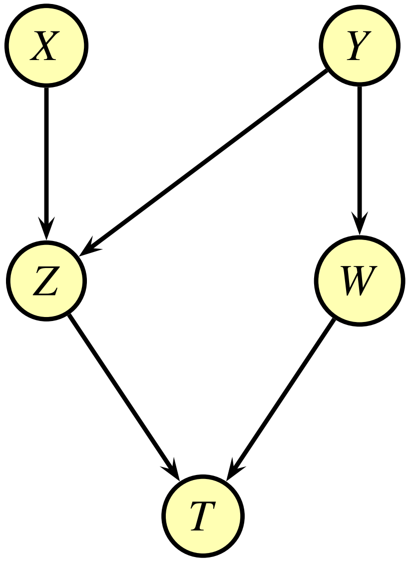


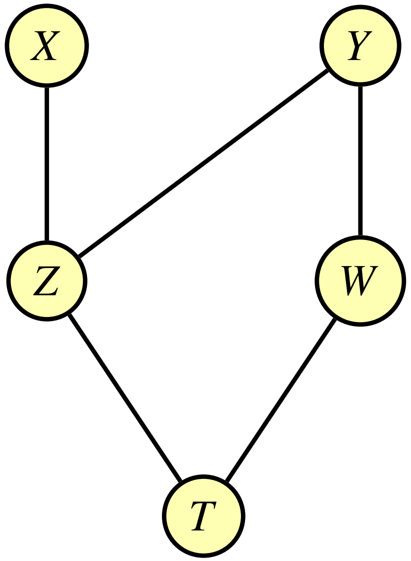


YES

$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

Example

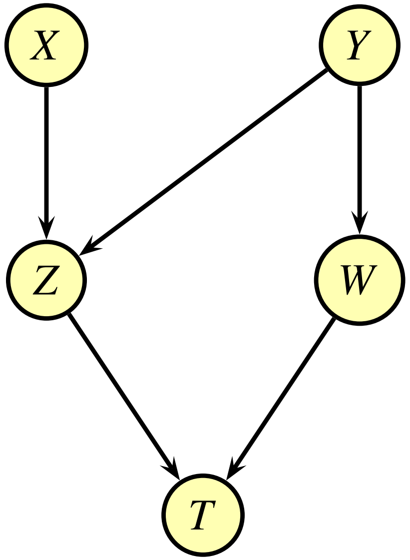


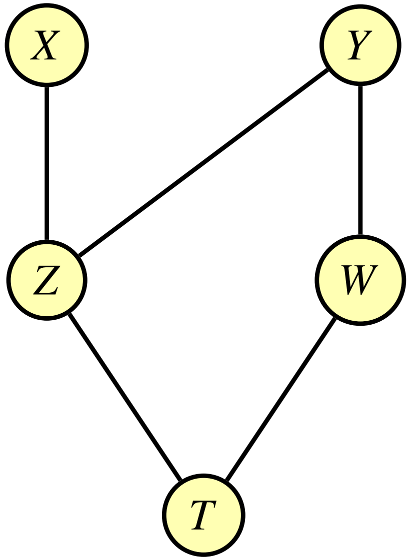


YES

$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z, W\}$$
$$S_{Y,T} = \{Z, W\}$$

Example





$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z, W\}$$
$$S_{Y,T} = \{Z, W\}$$

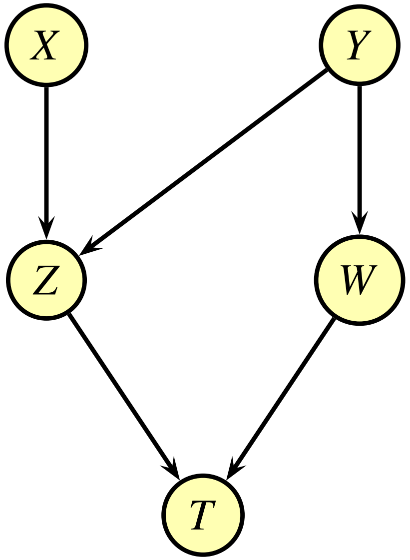
We stop because there is no node with more than 3 adjacent nodes

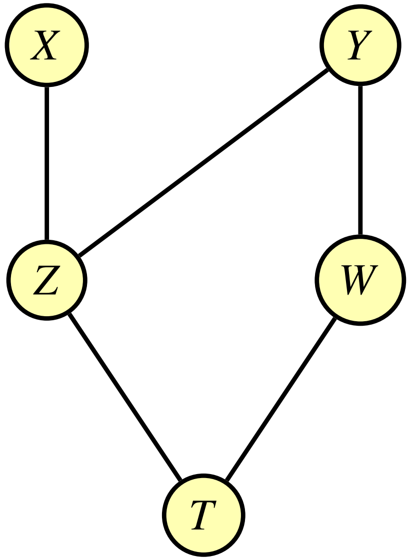
Finding Head-to-Head Links

1. **For each** uncoupled meeting $X - Z - Y$
   2. **If** $Z \notin S_{XY}$
      3. Orient $X - Z - Y$ as $X \rightarrow Z \leftarrow Y$

If a variable is connected with other two variables and it is not in the separator of them, then the arrows have to be head-to-head.

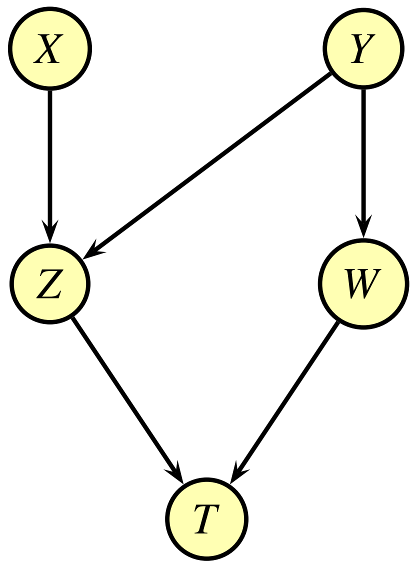
Example

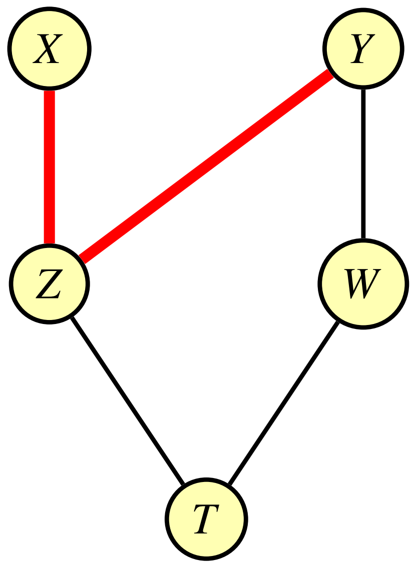




$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

Example

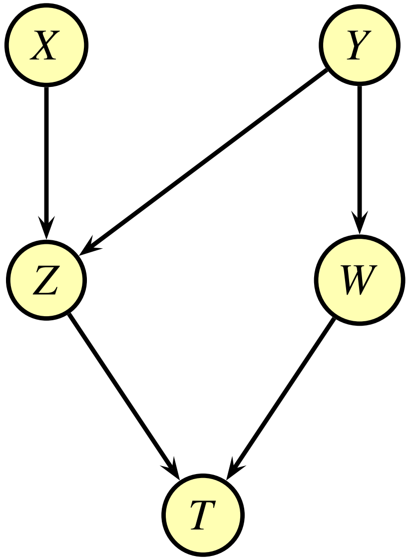


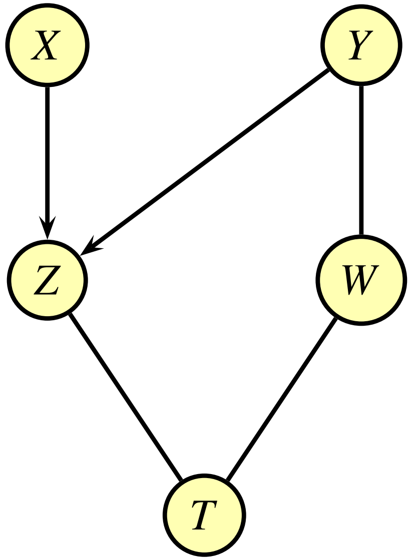


$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

$$Z \notin S_{X,Y}$$

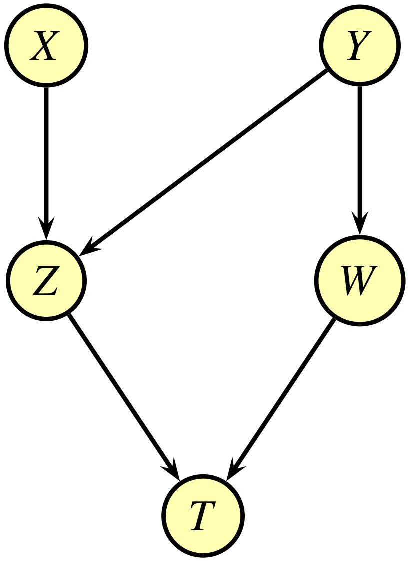
Example

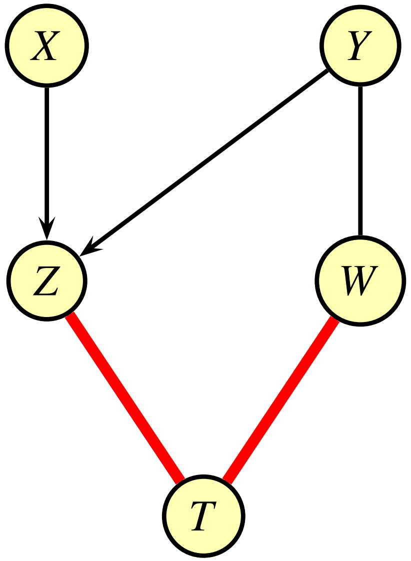




$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

Example

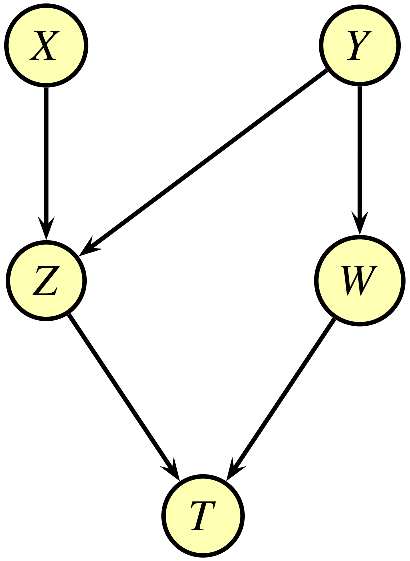


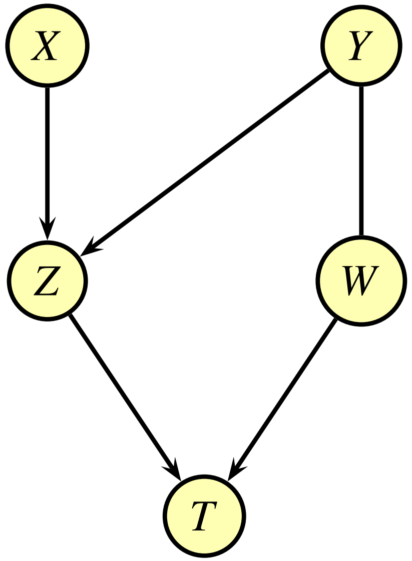


$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

$$T \notin S_{Z,W}$$

Example





$$S_{X,Y} = \emptyset$$
$$S_{X,W} = \emptyset$$
$$S_{Z,W} = \{Y\}$$
$$S_{X,T} = \{Z,W\}$$
$$S_{Y,T} = \{Z,W\}$$

This finishes the initial partial orientation

More Orientations

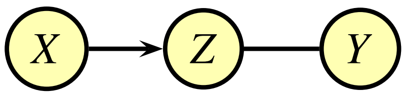
The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \rightarrow Z - Y$
      3. Orient $Z - Y$ as $Z \rightarrow Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \rightarrow Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \rightarrow W, Y \rightarrow W, Z - W$
      7. Orient $Z - W$ as $Z \rightarrow YW$

More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

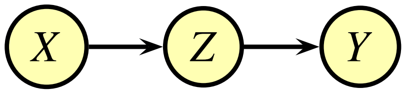
1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \rightarrow Z - Y$
      3. Orient $Z - Y$ as $Z \rightarrow Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \rightarrow Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \rightarrow W, Y \rightarrow W, Z - W$
      7. Orient $Z - W$ as $Z \rightarrow YW$



More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

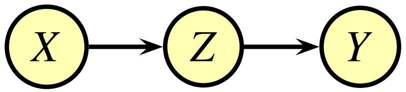
1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \rightarrow Z - Y$
      3. Orient $Z - Y$ as $Z \rightarrow Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \rightarrow Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \rightarrow W, Y \rightarrow W, Z - W$
      7. Orient $Z - W$ as $Z \rightarrow YW$

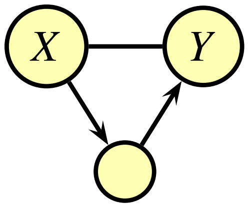


More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \to Z - Y$
      3. Orient $Z - Y$ as $Z \to Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \to Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \to W, Y \to W, Z - W$
      7. Orient $Z - W$ as $Z \to YW$

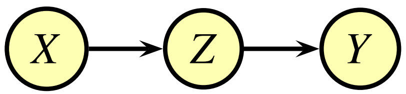


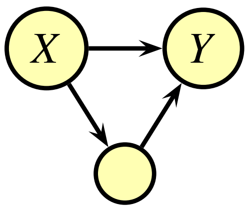


More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \rightarrow Z - Y$
      3. Orient $Z - Y$ as $Z \rightarrow Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \rightarrow Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \rightarrow W, Y \rightarrow W, Z - W$
      7. Orient $Z - W$ as $Z \rightarrow YW$

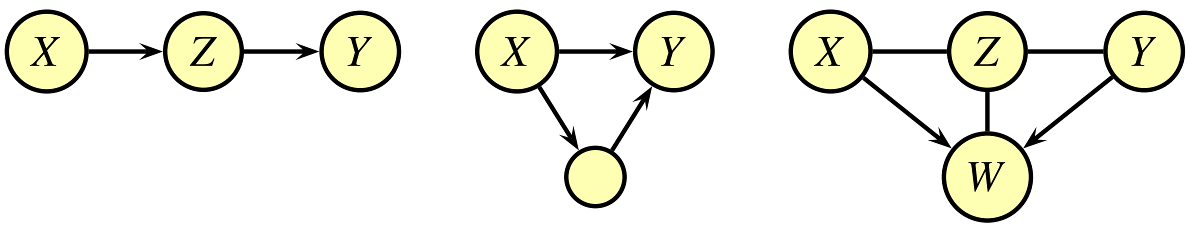




More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

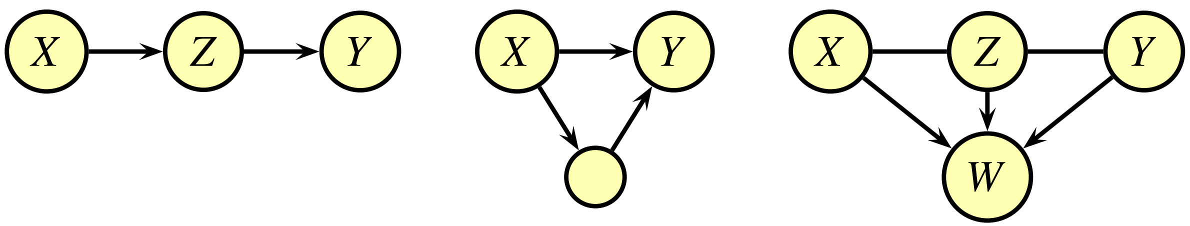
1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \to Z - Y$
      3. Orient $Z - Y$ as $Z \to Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \to Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \to W, Y \to W, Z - W$
      7. Orient $Z - W$ as $Z \to YW$



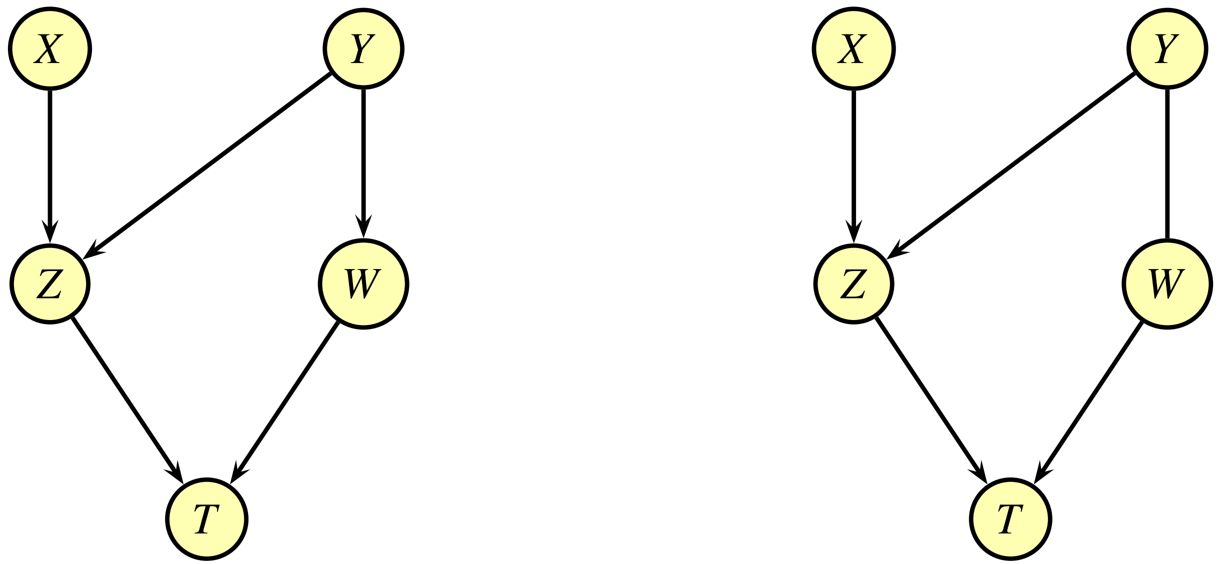
More Orientations

The basic idea is that no new head-to-head links are created and that the DAG condition is preserved.

1. **While** no more edges can be oriented
   2. **For** each uncoupled meeting $X \rightarrow Z - Y$
      3. Orient $Z - Y$ as $Z \rightarrow Y$
   4. **For** each $X - Z$ such that there is a directed path from X to Z
      5. Orient $X - Y$ as $X \rightarrow Y$
   6. **For** each uncoupled meeting $X - Z - Y$ such that $X \rightarrow W, Y \rightarrow W, Z - W$
      7. Orient $Z - W$ as $Z \rightarrow YW$

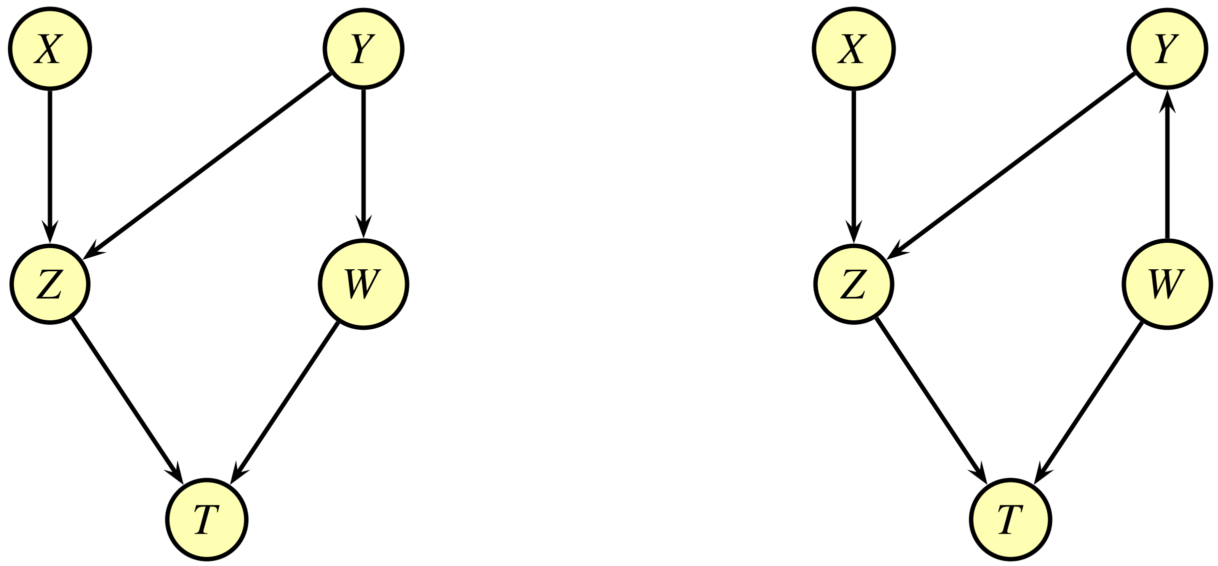


Example



The rest of arcs are oriented on an arbitrary way, keeping the DAG conditioning and not creating head-to-head links.
This may be a cause of counterintuitive head to head links.

Example

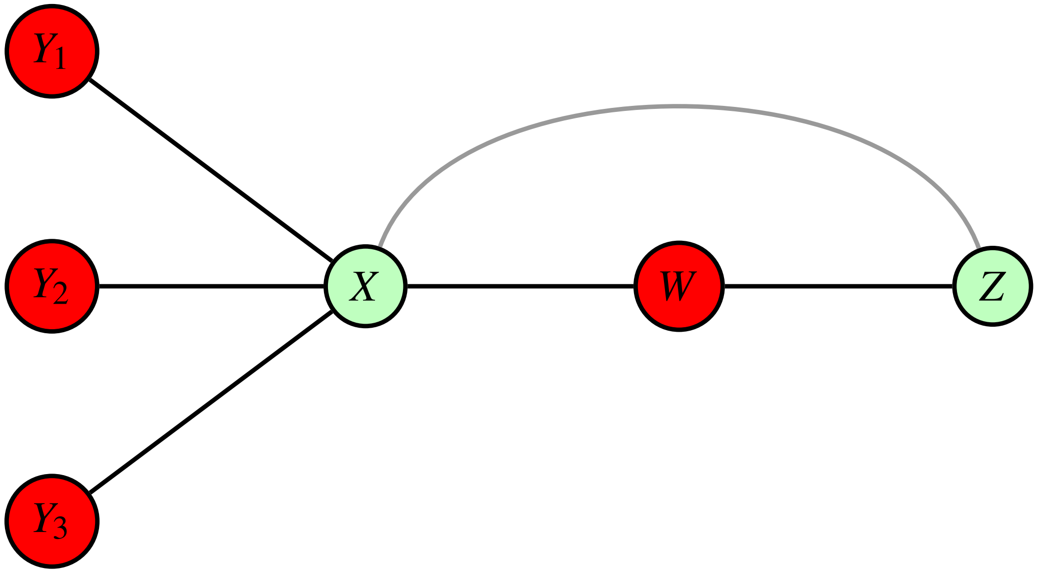


The rest of arcs are oriented on an arbitrary way, keeping the DAG conditioning and not creating head-to-head links.
This may be a cause of counterintuitive head to head links.

Problems of PC Algorithm

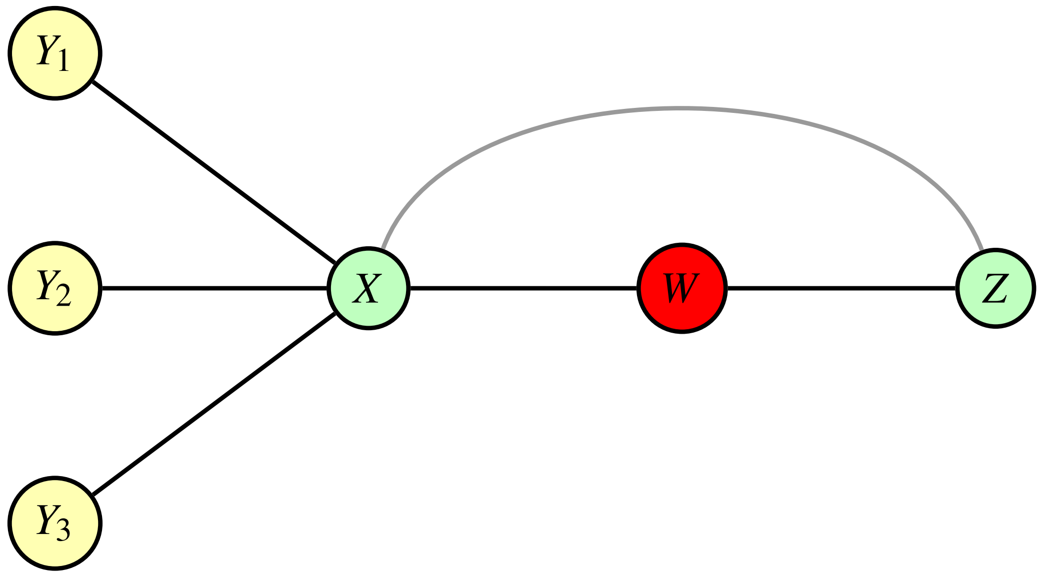
- The algorithm assumes that the set of independences can be exactly represented by a DAG.
  Example.- 3 variables pairwise marginally independent, and each one is conditionally dependent of the other, given the third. PC learns an empty network.
- There are too many tests and some of them may fail. This is specially important if the sample is small or we are conditining to a large set of variables: the tests have a tendency to produce independence.

Design Problems



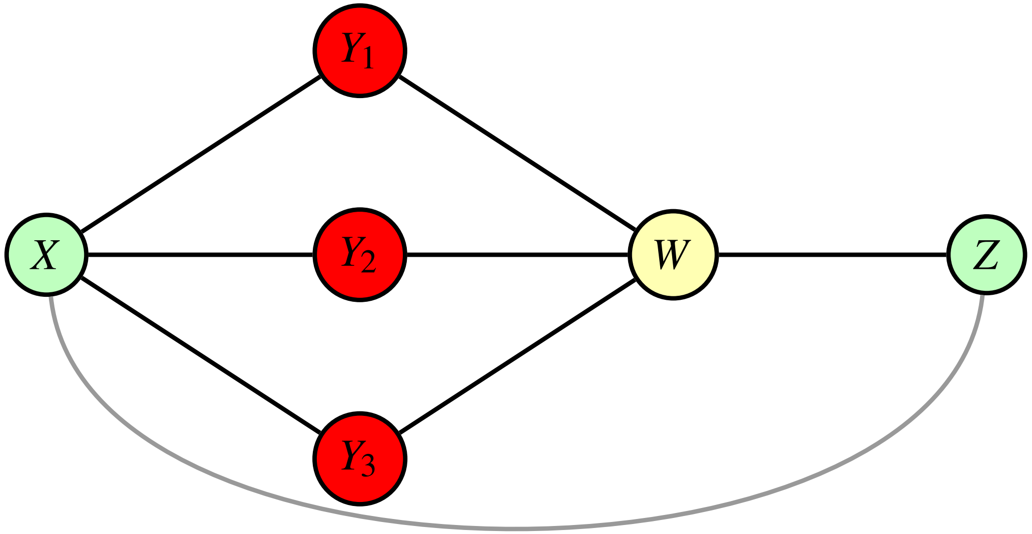
If we are tryint to test the link $X - Y$ then we make an independence test given a subset of neighbouring nodes to one of them (for example X). Only W would be necessary.
This produces a tendency to delete arcs.

NPC Algorithm (Hugin)



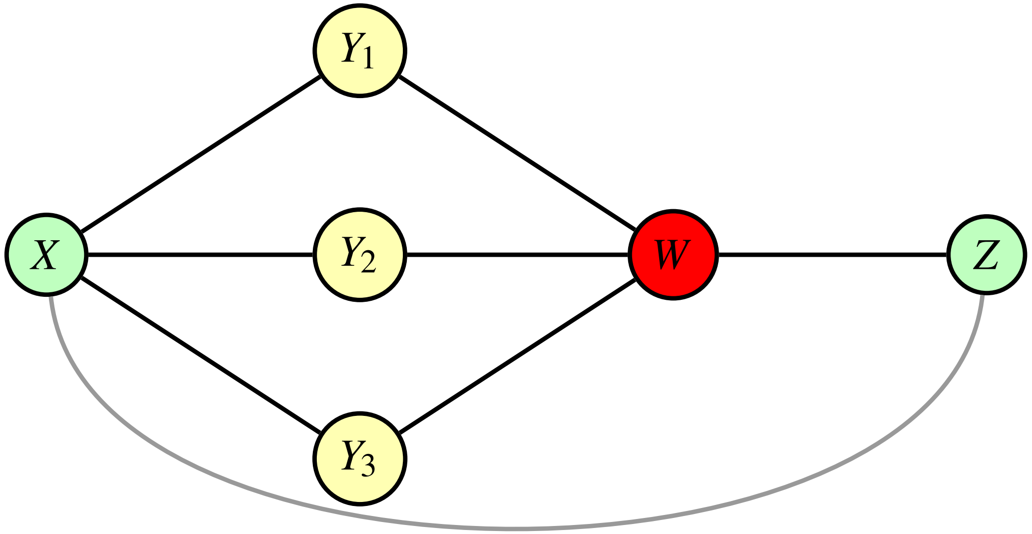
It only considers the nodes in the path from X to Z.

Some Remaining Problems



Again we are conditioning to a too large set $\{Y_1, Y_2, Y_3\}$. Only W would be enough.

BPC Algorithm



It is enough to make independence tests for all the subsets of a set of nodes separating X and Z.
In BPC we consider the subsets of a minimum cut set between X and Z. So we minimize the number of tests and the size of the conditioning set.

Statistical Tests

The algorithm is based in asking for the true of independence relationships of the form:

$$I(X_i, X_j | A)$$

where A is a subset of variables.
It can work with any source providing this kind of information.
It we have a database, this is answered by means of statistical tests of independence.

Cross Entropy

If X, Y are random variables with joint probability distribution P the cross entropy between them is

$$CE(X,Y) = \sum_{x,y} P(x,y) \log \left(\frac{P(x,y)}{P(x).P(y)} \right)$$

It is a non-negative value, bounded by the entropy of X and Y. In fact, if $H(X)$ denotes the entropy of X, then

$$CE(X,Y) = H(X) - H(X|Y)$$

where $H(X|Y)$ is the entropy of X conditioned to Y:
$H(X|Y) = \sum_y H(H|y)P(y)$
It is 0.0 only for independent variables and measures the degree of dependence of two variables (any kind of relation).

The Conditional Cross Entropy

Given three variables X, Y, Z the cross entropy of X and Y given Z is defined as

$$CE(X,Y|Z) = \sum_z P(z) \sum_{x,y} P(x,y|z) \log\left(\frac{P(x,y|z)}{P(x|z).P(y|z)}\right)$$

It verifies the $CE(X,Y|Z) = H(X|Z) - H(X|Y,Z)$
This value is also called the Mutual Information.
I can be analogously defined when Z is a set of variables.

It verifies similar properties to unconditional entropy. It measures the degree of dependence of X and Y given Z. In particular, it is equal to 0.0 when this conditional independence is verified.

Independence Test

To test whether X and Y are conditional independent given A, we compute the cross entropy $CE(X,Y|A)$ where the probabilities are their maximum likelihood estimators from the database (relative frequencies).
The statistic used for the test is G^2 which is $2mCE(X,Y|A)$ where m is the sample size.
It is known that, under the independence assumption, G^2 follows a χ^2 distribution with degrees of freedom equal to:

$$(r_X - 1)(r_Y - 1)\prod_{Z \in A} r_Z$$

where r_W is the number of values of variable W.

Sample Size

- The significance level is usually 0.01 or 0.05 (greater values are better).
- When sample size or the conditional set is big, then the possibility of rejecting the null hypothesis is lower and independence will be assumed: Lack of support implies independence

Results

Dependence

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	10000	8119	8119	5008	5008	8119	6879	8119
5	8292	6787	6787	5149	5149	6787	5848	6787
10	7306	5635	5521	4893	4667	6001	4582	5868
20	5788	4234	4627	4294	3927	5406	3525	4609
50	4222	3021	3693	3489	3141	4256	2502	3400
100	3180	2213	2920	2853	2527	3445	1790	2550
1000	1138	763	1307	1238	1123	1515	610	1000
10000	346	224	480	465	429	542	174	324

Results

Independence

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	0	841	841	3322	3322	841	2485	841
5	335	1548	1548	2622	2622	1548	2920	1548
10	389	1698	1912	2030	2300	1304	3223	2082
20	499	2057	1643	1559	1992	773	3290	2441
50	515	2100	1199	1053	1585	497	3310	2644
100	522	2099	904	758	1208	318	3303	2678
1000	536	2098	299	236	399	90	3259	2754
10000	517	1968	89	64	117	21	3182	2705

Results

Total Errors

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	10000	8960	8960	8330	8330	8960	9364	8960
5	8627	8335	8335	7771	7771	8335	8768	8335
10	7695	7333	7433	6923	6967	7305	7805	7950
20	6287	6291	6270	5853	5919	6179	6815	7050
50	4737	5121	4892	4542	4726	4753	5812	6044
100	3702	4312	3824	3611	3735	3763	5093	5228
1000	1674	2861	1606	1474	1522	1605	3869	3754
10000	863	2192	569	529	546	563	3356	3029

Results

Statistical Tests

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	10000	8960	8960	8330	8330	8960	9364	8960
5	8627	8335	8335	7771	7771	8335	8768	8335
10	7695	7333	7433	6923	6967	7305	7805	7950
20	6287	6291	6270	5853	5919	6179	6815	7050
50	4737	5121	4892	4542	4726	4753	5812	6044
100	3702	4312	3824	3611	3735	3763	5093	5228
1000	1674	2861	1606	1474	1522	1605	3869	3754
10000	863	2192	569	529	546	563	3356	3029

Statistical tests should decrease significance level with the sample size in order to minimize the total number of errors.

Results - K-L distance

Dependence

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	0.28794	0.25469	0.25469	0.23437	0.23437	0.25469	0.25134	0.25469
5	0.19904	0.18584	0.18584	0.17455	0.17455	0.18584	0.18319	0.18584
10	0.12943	0.11676	0.11620	0.11400	0.11287	0.11865	0.11402	0.12070
20	0.07050	0.06412	0.06520	0.06409	0.06327	0.06850	0.06279	0.06666
50	0.03032	0.02778	0.02898	0.02858	0.02816	0.03045	0.02724	0.02904
100	0.01551	0.01438	0.01510	0.01514	0.01472	0.01603	0.01417	0.01503
1000	0.00155	0.00150	0.00160	0.00157	0.00155	0.00165	0.00149	0.00155
10000	0.00015	0.00015	0.00016	0.00016	0.00016	0.00016	0.00015	0.00015

Again Akaike is very good (if 'a priori' dependence)

Results - K-L distance

Independence

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	0.17470	0.19164	0.19164	0.20577	0.20577	0.19164	0.19723	0.19164
5	0.13717	0.14940	0.14940	0.15553	0.15553	0.14940	0.15339	0.14940
10	0.08367	0.09341	0.09398	0.09312	0.09440	0.09127	0.09709	0.09175
20	0.04820	0.05427	0.05312	0.05227	0.05327	0.04982	0.05653	0.05330
50	0.02145	0.02449	0.02297	0.02253	0.02324	0.02139	0.02560	0.02397
100	0.01104	0.01271	0.01150	0.01122	0.01163	0.01065	0.01333	0.01241
1000	0.00115	0.00134	0.00109	0.00108	0.00111	0.00104	0.00141	0.00131
10000	0.00011	0.00013	0.00010	0.00010	0.00010	0.00010	0.00014	0.00013

Results - K-L distance

Total Error

Size	$CHI^{0.05}$	$CHI^{0.20}$	K2	BD2	BD4	BIC	AKA	IMP
3	0.46264	0.44632	0.44633	0.440141	0.440141	0.446327	0.448566	0.446327
5	0.33621	0.33523	0.33524	0.330078	0.330078	0.335237	0.336579	0.335237
10	0.21309	0.21016	0.21018	0.207127	0.207272	0.209917	0.211113	0.212449
20	0.11870	0.11838	0.11831	0.116357	0.116538	0.118318	0.119318	0.119957
50	0.05177	0.05226	0.05195	0.051115	0.051396	0.051842	0.052843	0.05301
10^2	0.02655	0.02709	0.02660	0.026362	0.026347	0.026679	0.027503	0.027445
10^3	0.0027	0.00283	0.00269	0.002655	0.002663	0.002688	0.002897	0.002864
10^4	0.00026	0.00028	0.00026	0.000257	0.000256	0.000259	0.000291	0.000284

BD2 is the best, and IMP is bad for intermediate samples, AKA bad for large samples

Bayesian Equivalent Scores

$$P(D|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(s_{ij})}{\Gamma(N_{ij}+s_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}$$

- Independence $I(X_i, X_j|A)$: Compute the score for X_i with A the parents of X_i:

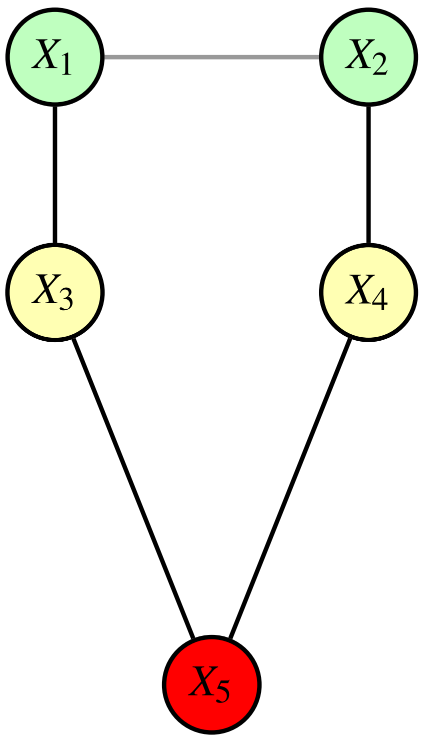
$$C = \prod_{j=1}^{q_i} \frac{\Gamma(s_{ij})}{\Gamma(N_{ij}+s_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})}$$

- Dependence $I(X_i, X_j|A)$: Compute the score for X_i with $A \cup \{Y\}$ the parents of X_i:

Compare the two scores to decide.

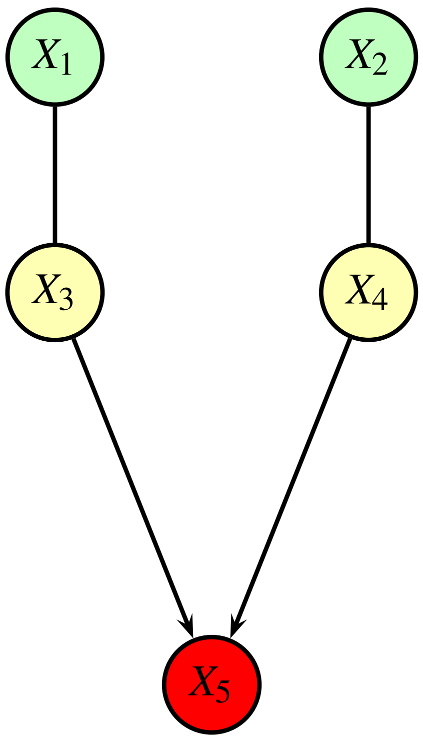
Refinement

Some of the decisions taken at first stages of the algorithm are not appropriate afterward. For example, assume:



Refinement

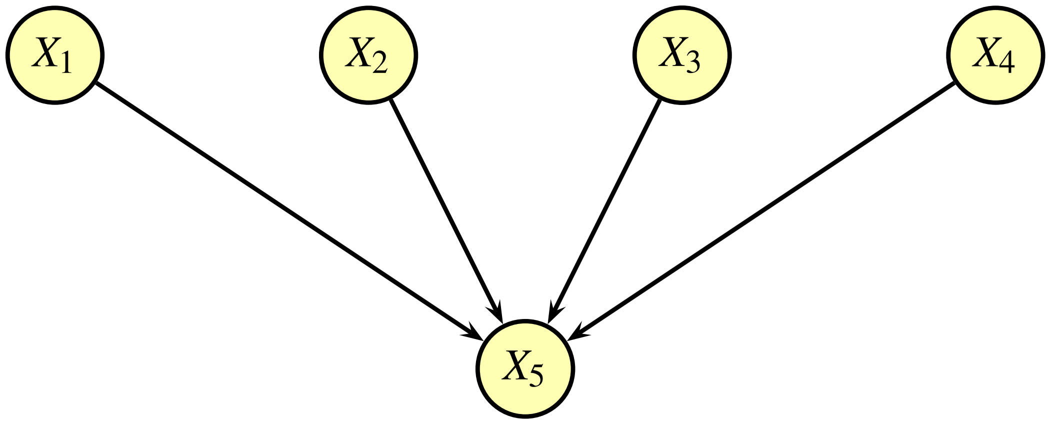
Some of the decisions taken at first stages of the algorithm are not appropriate afterward. For example, assume:



And we make afterwards the orientation. It can be convinient to add arcs.

Refinement

Even if we have been able of determining the correct structure, if we want to approximate the joint probability it can be more convenient to simplify the graph



We have carried out a local hill-climbing search to improve the score. Starting with a total order compatible with the graph and then adding or deleting arcs compatible with this order.

Asia

- Without refinement, BPC adds more, but removes much less than PC. Less total errors.
- BPC produces lower KL distance
- Scores are better
- Refinement increases the number of errors but decreases KL distance

Alarm

- Without refinement, BPC adds more, but removes much less than PC. Less total errors.
- BPC produces lower KL distance in some cases. PC produces lower KL distance with large samples, scores, and refinement. PC has a tendency to be better than BPC (KL distance) with larger samples, refinement and scores.
- Scores are better
- Refinement increases the number of errors but decreases KL distance
- Times are better for PC, except for very large samples.