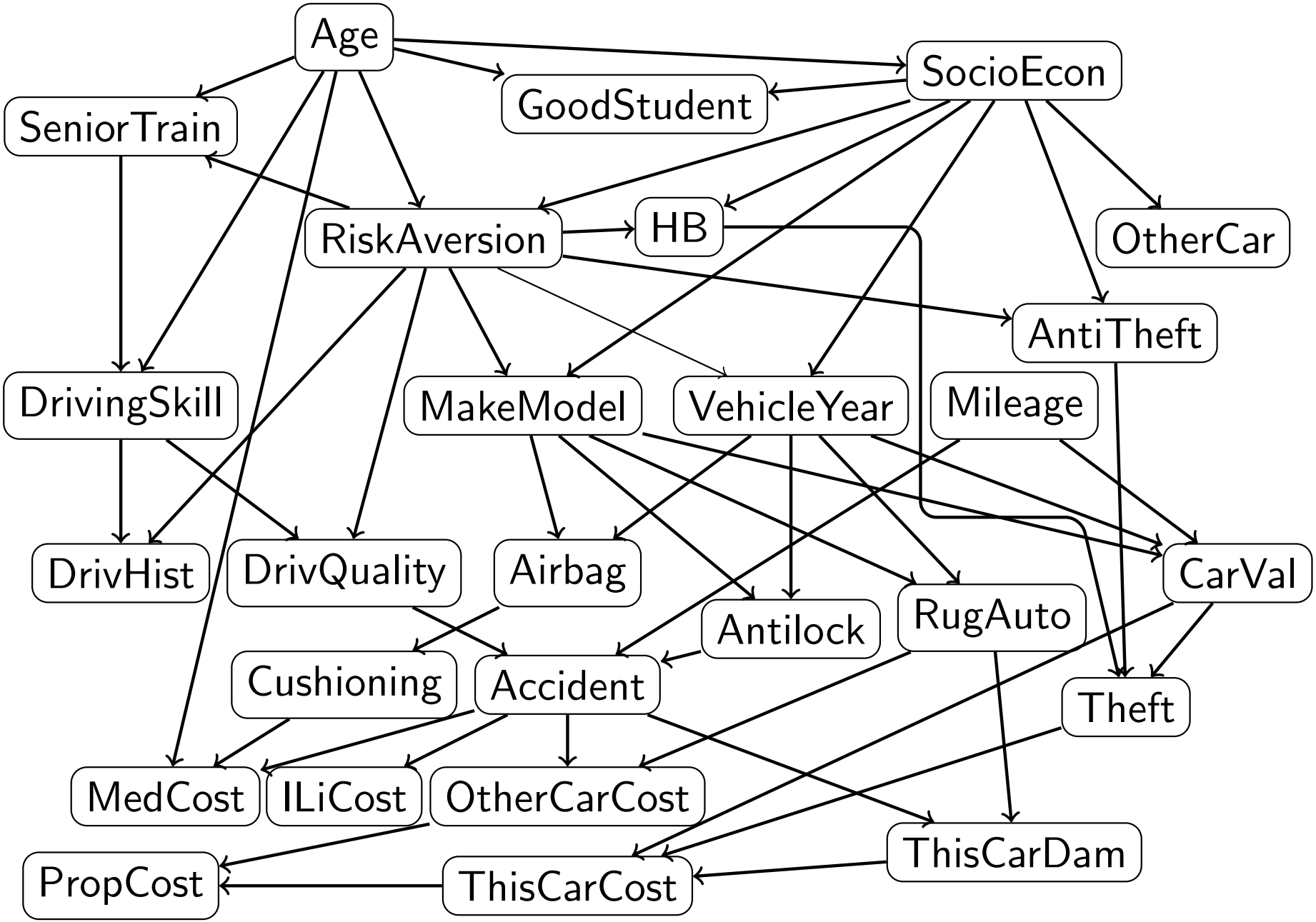
# Bayesian network model selection using integer programming

James Cussens, University of York

Oxford, 2015-06-04


Age
SocioEcon
GoodStudent
SeniorTrain
RiskAversion
HB
OtherCar
AntiTheft
DrivingSkill
MakeModel
VehicleYear
Mileage
DrivHist
DrivQuality
Airbag
CarVal
Antilock
RugAuto
Cushioning
Accident
Theft
MedCost
ILiCost
OtherCarCost
ThisCarDam
PropCost
ThisCarCost

# The BDeu score

Given complete discrete data $D$, with an appropriate choice of Dirichlet priors for the parameters, the log marginal likelihood for BN structure $G$ with variables $i = 1, \ldots, p$ is:

$$\log P(D|G) = \sum_{i=1}^{p} c_i(G)$$

where

$$c_i(G) = c_{i \leftarrow \mathrm{Pa}_G(i)} = \sum_{j=1}^{q_i(G)} \left( \log \frac{\Gamma(\alpha_{ij})}{\Gamma(n_{ij} + \alpha_{ij})} + \sum_{k=1}^{r_i} \log \frac{\Gamma(n_{ijk} + \alpha_{ijk})}{\Gamma(\alpha_{ijk})} \right)$$

depends only on the parents variable $i$ has in graph $G$.

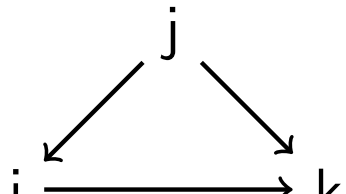
# Combinatorial optimisation

With the preceding assumptions the BN model selection problem is to find a $\check{G}$ such that:

$$\check{G} = \arg\max_G \left[\log P(D|G)\right] = \arg\max_G \left[\sum_{i=1}^{p} c_{i \leftarrow \mathrm{Pa}_G(i)}\right]$$

- This is a problem of *combinatorial optimisation*,
- which is known to be NP-hard.

# Encoding digraphs as real vectors

- The key to the integer programming (IP) approach to BN model selection is to view digraphs as points in $\mathbb{R}^n$.
- We do this via *family variables*.
- This digraph: 

is this point in $\mathbb{R}^{12}$:

| $i \leftarrow \{\}$ | $i \leftarrow \{j\}$ | $i \leftarrow \{k\}$ | $i \leftarrow \{j,k\}$ |
|---|---|---|---|
| 0 | 1 | 0 | 0 |
| $j \leftarrow \{\}$ | $j \leftarrow \{i\}$ | $j \leftarrow \{k\}$ | $j \leftarrow \{i,k\}$ |
| 1 | 0 | 0 | 0 |
| $k \leftarrow \{\}$ | $k \leftarrow \{i\}$ | $k \leftarrow \{j\}$ | $k \leftarrow \{i,j\}$ |
| 0 | 0 | 0 | 1 |

# BDeu scores as linear objective

Let $x(G)$ be the vector for digraph $G$, then

$$\log P(D|G) = \sum_{i=1}^{p} c_{i \leftarrow \mathrm{Pa}_G(i)} = \sum_{i=1}^{p} \sum_{J : i \notin J} c_{i \leftarrow J} x(G)_{i \leftarrow J}$$

The optimisation problem then becomes: find $\check{x}$ such that

1. $\check{x} = \arg\max cx$
2. and $\check{x}$ represents an acyclic digraph.

# The integer program

We can ensure that $x$ represents an acyclic digraph with two classes of linear constraints and an integrality constraint.

1. 'convexity' $\forall i : \sum_J x_{i \leftarrow J} = 1$
2. 'cluster' $\forall C : \sum_{i \in C} \sum_{J \cap C = \emptyset} x_{i \leftarrow J} \geq 1$
3. $x$ is a zero-one vector

We have an *integer program*: $\max cx$ subject to the above constraints. It is an IP since:

- the objective function is linear
- there are only linear and integrality constraints

# Relaxation

Solving the following *relaxation* of the problem is very easy

1. $\forall i : \sum_J x_{i \leftarrow J} = 1$
2. ~~$\forall C : \sum_{i \in C} \sum_{J \cap C = \emptyset} x_{i \leftarrow J} \geq 1$~~ (combinatorial relaxation)
3. ~~$x$ is a zero-one vector~~ (linear relaxation)

Relaxations:

- provide an upper bound on an optimal solution,
- and we might 'get lucky' and find that the solution to the relaxation satisfies all the constraints of the original problem.

# Tightening the relaxation

- We tighten the relaxation by adding *cutting planes*
- Let $x^*$ be the solution to the current relaxation,
- If $\sum_{i \in C} \sum_{J \cap C = \emptyset} x^*_{i \leftarrow J} < 1$ then the valid inequality $\sum_{i \in C} \sum_{J \cap C = \emptyset} x_{i \leftarrow J} \geq 1$ is added to get a new relaxation,
- and so on.

- This procedure improves the upper bound.
- We might get lucky and find that $x^*$ represents an acyclic digraph, in which case the problem is solved.
- The SCIP system will find additional non-problem-specific cutting planes as well.

# The separation problem

The *separation problem* is:

- Given $x^*$ (the solution to the current LP relaxation),
- Find $C$ such that $\sum_{i \in C} \sum_{J \cap C = \emptyset} x^*_{i \leftarrow J} < 1$, or show that no such $C$ exists.

- This separation problem has recently been shown to be NP-hard [CJKB15].
- In the GOBNILP system a sub-IP is used to solve it.
- Note: the vast majority of cluster inequalities are **not** added, since they do not tighten the relaxation.

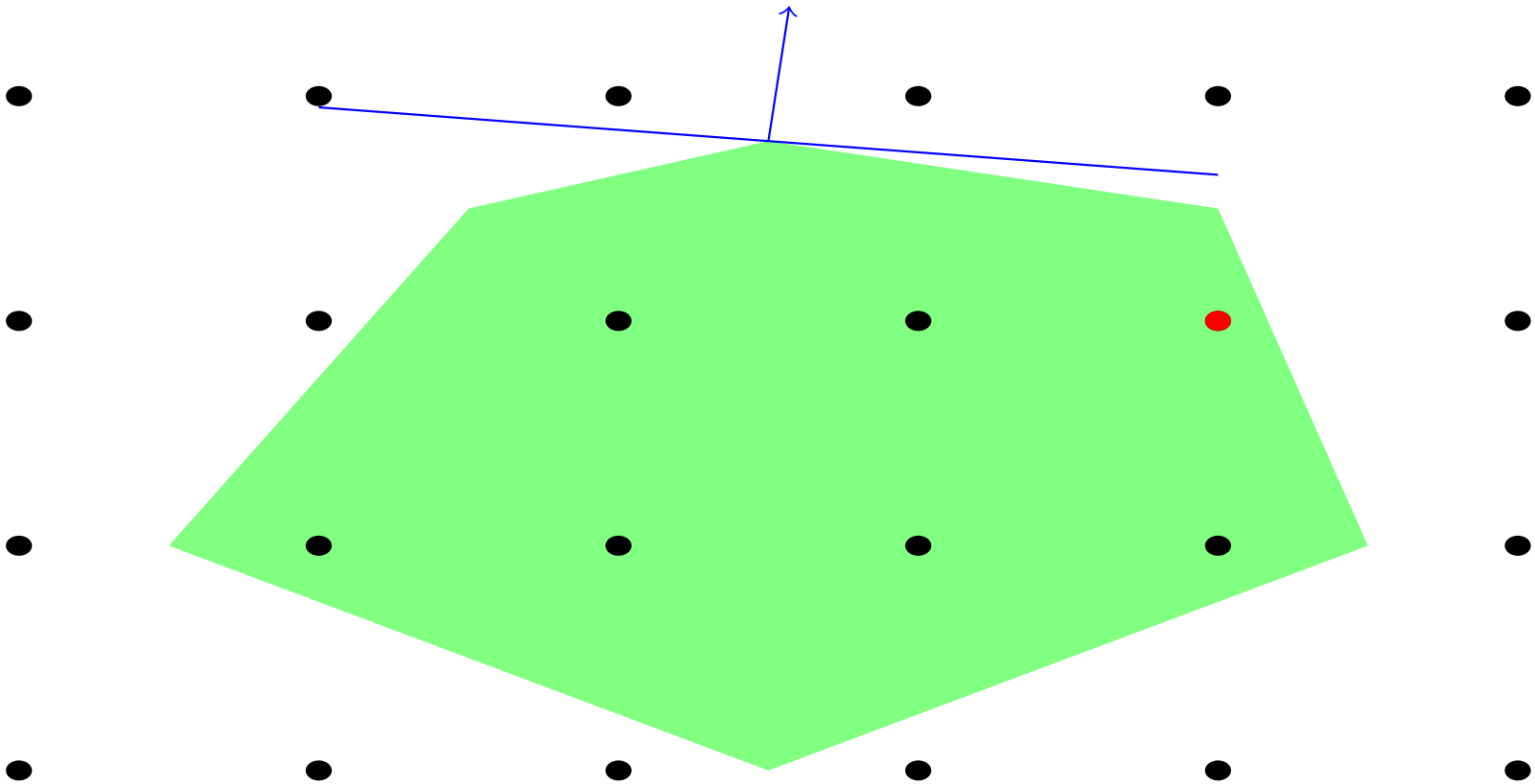
# Getting lucky . . . eventually

Eskimo pedigree. 1614 BN variables. At most 2 parents. Simulated genotypes. 11934 IP variables.

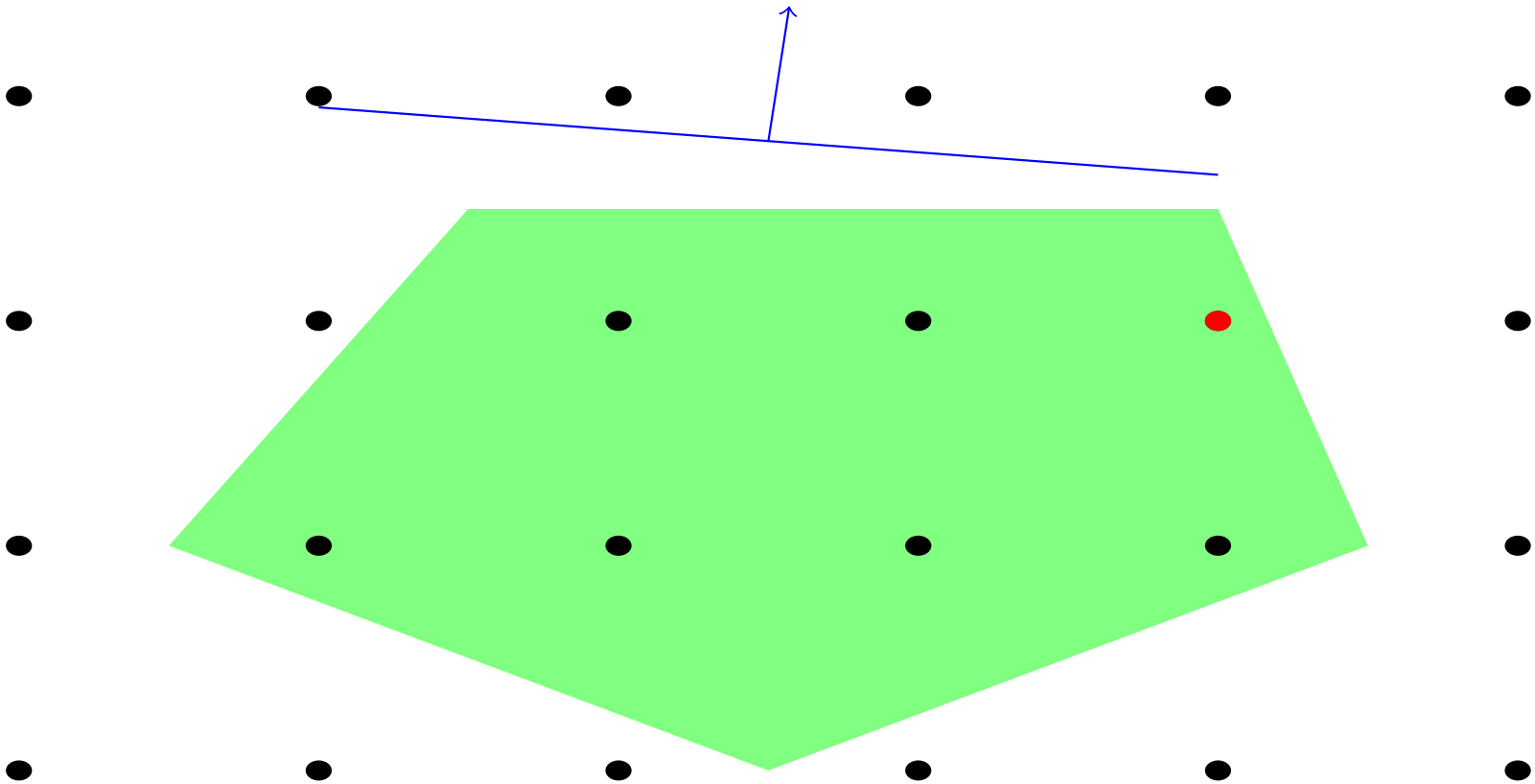
```
 time |frac|cuts |   dualbound   | primalbound   |  gap
 1110s|120 | 661 | -3.162149e+04 |-4.616035e+04 |  45.98%
 1139s|118 | 669 | -3.162175e+04 |-4.616035e+04 |  45.98%
 1171s| 94 | 678 | -3.162213e+04 |-4.616035e+04 |  45.97%
 1209s| 26 | 684 | -3.162220e+04 |-4.616035e+04 |  45.97%
 1228s|103 | 685 | -3.162223e+04 |-4.616035e+04 |  45.97%
 1264s|  0 | 692 | -3.162234e+04 |-4.616035e+04 |  45.97%
*1266s|  0 |   - | -3.162234e+04 |-3.162234e+04 |   0.00%

SCIP Status        : problem is solved [optimal solution found
Solving Time (sec) : 1266.40
```

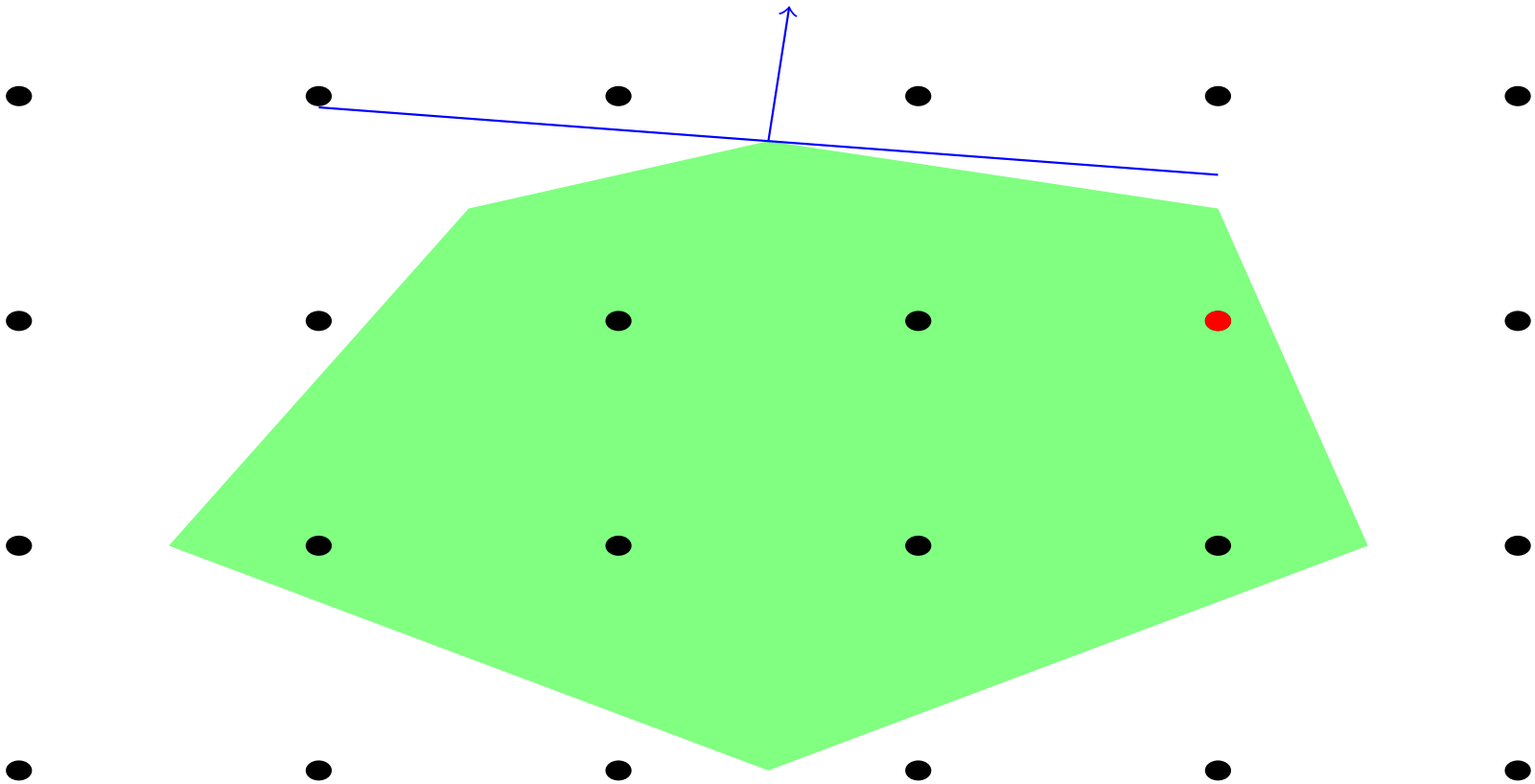
# Cutting planes from integrality constraints



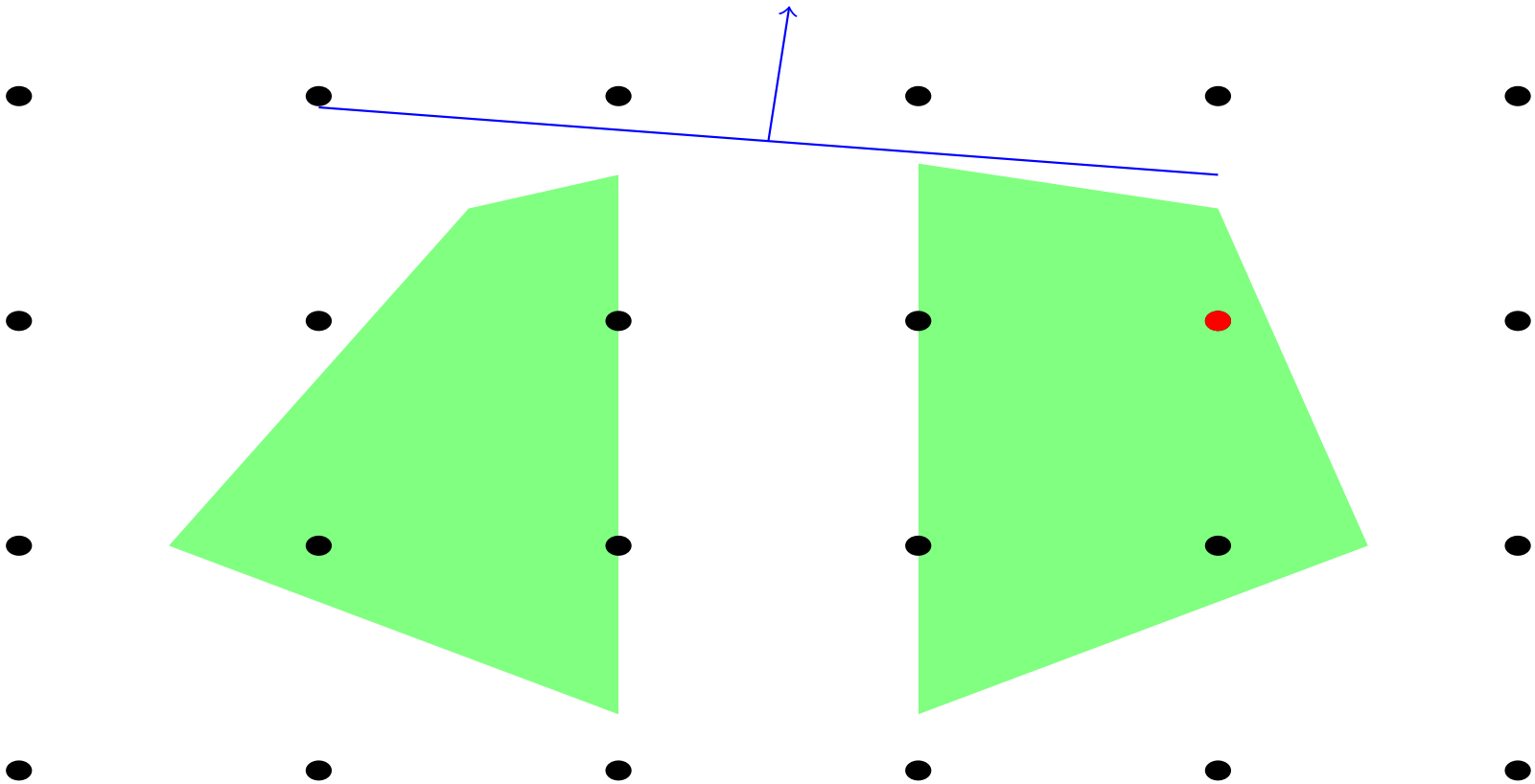
# Cutting planes from integrality constraints



# Branch-and-bound



# Branch-and-bound



# Branch and cut

```
1. Let x* be the LP solution.
2. If x* worse than incumbent then exit.
3. If there are valid inequalities
        not satisfied by x*
        add them and go to 1.
   Else if x* is integer-valued then
        the current problem is solved
   Else branch on a variable with
        non-integer value in x*
        to create two new sub-problems
        (propagate if possible)
```

# The convex hull

- Since each acyclic digraph is a point in $\mathbb{R}^n$ there is a convex hull of acyclic digraphs.
- If our IP had all the inequalities defining this convex hull we could drop the integrality restriction and solve the problem with a *linear program* (LP).
- An LP, unlike, an IP, can be solved in polynomial time.
- For 4 BN variables, there are 543 acyclic digraphs (living in $\mathbb{R}^{28}$) and the convex hull is defined by 135 inequalities.

# Facets

- The inequalities defining the convex hull are called *facets*.
- We have shown [CJKB15, CHS15] that the cluster inequalities, first introduced by [JSGM10], are facets.
- But there are very many other facets, for example this one for BN variable set $\{a, b, c, d\}$:

$$\begin{aligned} & x_{a\leftarrow bc} + x_{a\leftarrow bd} + x_{a\leftarrow cd} + 2x_{a\leftarrow bcd} \\ + & x_{b\leftarrow ac} + x_{b\leftarrow ad} + x_{b\leftarrow acd} \\ + & x_{c\leftarrow ab} + x_{c\leftarrow ad} + x_{c\leftarrow abd} \\ + & x_{d\leftarrow ab} + x_{d\leftarrow ac} + x_{d\leftarrow abc} \qquad \leq 2 \end{aligned}$$

# Characteristic imsets and matroids

- An alternative approach—*characteristic imsets*, developed by Milan Studený—encodes each Markov equivalence class of BNs as a zero-one vector [CHS15].
- There is a (non-injective) linear map between family-variable vectors and c-imsets.
- Studený has recently used matroid theory to derive useful results for both the c-imset and family-variable polytope [Stu15].

# Limitations

- Since the BN model selection problem is NP-hard (even with convenient assumptions such as complete data),
- all methods have to compromise in some way.
- If the IP solving terminates then we have a BN with guaranteed maximal marginal likelihood,
- but if we need too many IP variables then we may not get any solution, due to running out of memory.
- For most problems, it is possible to show that very many family variables will have value zero in any optimal solution—this is what saves us.
- And/or we can limit the size of parent sets.

# Are global optima worth the effort?

- In a recent paper, Malone *et al* [MJM15] show that "exact approaches, which guarantee to find globally optimal solutions, consistently generalize well to unseen testing data,"
- In our work using synthetic genetic data, we [SBC14] too found that 'optimal' pedigrees were (modestly) more accurate than those found by a heuristic algorithm.

# Extensions

- If we have prior knowledge, such as conditional independence relations, we add these as additional constraints.
- If the prior on BN structures is 'modular' then we can do MAP model selection (the uniform prior is trivially modular).
- We can add constraints to rule out *immoralities* to learn decomposable models.
- Oates *et al* [OSMC15] learned multiple BNs (from multiple datasets) with a penalty for structural differences.


James Cussens, David Haws, and Milan Studený.
Polyhedral aspects of score equivalence in Bayesian network structure learning.
Arkiv 1503.00829, March 2015.


James Cussens, Matti Järvisalo, Janne H. Korhonen, and Mark Bartlett.
Polyhedral theory for Bayesian network structure learning.
in preparation, June 2015.


Tommi Jaakkola, David Sontag, Amir Globerson, and Marina Meila.
Learning Bayesian network structure using LP relaxations.
In *Proceedings of 13th International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, volume 9, pages 358–365, 2010.
Journal of Machine Learning Research Workshop and Conference Proceedings.


Brandon Malone, Matti Järvisalo, and Petri Myllymäki.

Impact of learning strategies on the quality of Bayesian networks: An empirical evaluation.
In *Proc. UAI 2015*, 2015.


Chris Oates, Jim Smith, Sach Mukherjee, and James Cussens.
Exact estimation of multiple directed acyclic graphs.
*Statistics and Computing*, 2015.
Forthcoming.


Nuala Sheehan, Mark Bartlett, and James Cussens.
Improved maximum likelihood reconstruction of complex multi-generational pedigrees.
*Theoretical Population Biology*, 97:11–19, 2014.


Milan Studený.
How matroids occur in the context of learning Bayesian network structure.
In *Proc. UAI 2015*, 2015.