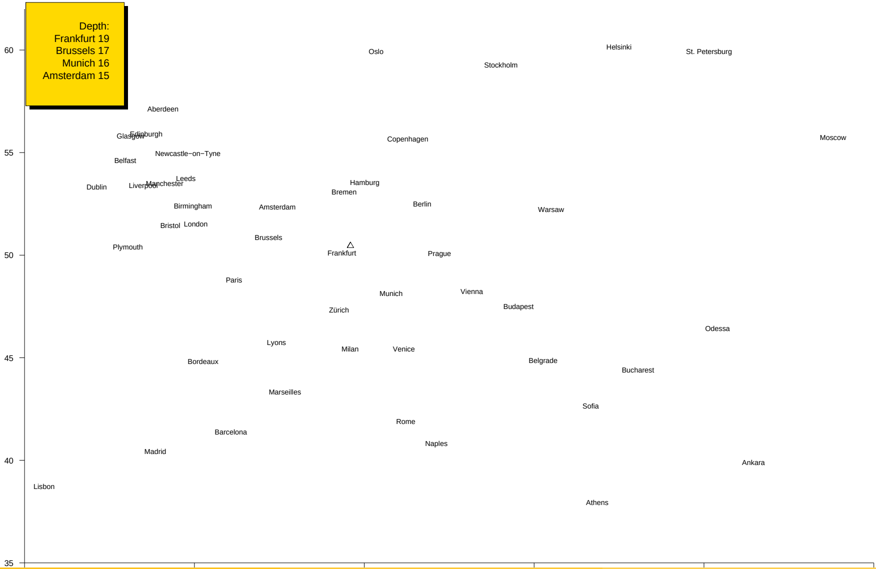
Halfspace depth: motivation, computation, optimization

David Bremner

UNB

March 27, 2007

Wir sind Zentrum



Robustness

- The *breakdown point* of an estimator is the fraction of data that must be moved to infinity before the estimator is also moved to infinity.
- The breakdown point of the mean is $\frac{1}{n}$ (i.e. one error suffices to destroy the estimate).
- The median in $\mathbb{R}^1$ has breakdown 1/2.

Robustness

- The *breakdown point* of an estimator is the fraction of data that must be moved to infinity before the estimator is also moved to infinity.
- The breakdown point of the mean is $\frac{1}{n}$ (i.e. one error suffices to destroy the estimate).
- The median in $\mathbb{R}^1$ has breakdown 1/2.

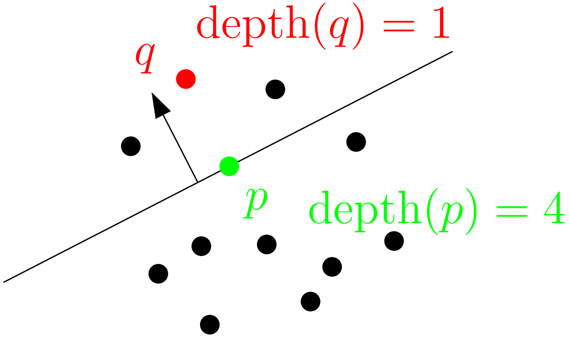
Robustness

- The *breakdown point* of an estimator is the fraction of data that must be moved to infinity before the estimator is also moved to infinity.
- The breakdown point of the mean is $\frac{1}{n}$ (i.e. one error suffices to destroy the estimate).
- The median in $\mathbb{R}^1$ has breakdown $1/2$.

Halfspace Depth

The *halfspace depth* of a point q with respect to $S \subset \mathbb{R}^d$ is defined as

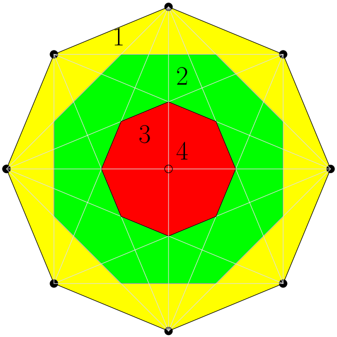
$$\text{depth}_S(q) = \min_{a \in \mathbb{R}^d \setminus 0} |\{ p \in S \mid \langle a, p \rangle \geq \langle a, q \rangle \}|$$



Halfspace Depth

The *halfspace depth* of a point q with respect to $S \subset \mathbb{R}^d$ is defined as

$$\mathrm{depth}_S(q) = \min_{a \in \mathbb{R}^d \setminus 0} |\{p \in S \mid \langle a, p \rangle \geq \langle a, q \rangle\}|$$

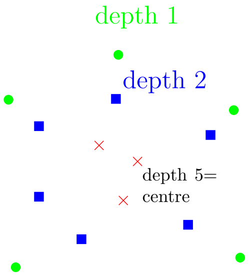


- Space is decomposed into nested convex regions of same depth

Tukey Median

The *Tukey Median* $t(S)$ is defined as

$$\{q \in S \mid \text{depth}_S(q) = \max_{p \in S} \text{depth}_S(p)\}$$

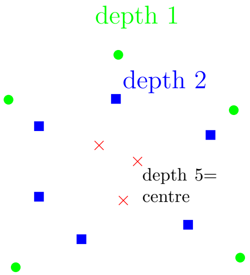


- The Tukey median has breakdown point at least $1/(d+1)$ for points in general position.

Tukey Median

The *Tukey Median* $t(S)$ is defined as

$$\{q \in S \mid \mathrm{depth}_S(q) = \max_{p \in S} \mathrm{depth}_S(p)\}$$



- The Tukey median has breakdown point at least $1/(d+1)$ for points in general position.

Depth of fit

- Statistical model with *parameters* $\vartheta = (\vartheta_1 \ldots \vartheta_p) \in \Theta$
- Datapoints Z
- *Criterial Functions* $F_z : \Theta \to [0, \infty)$, $z \in Z$

Definition

Model ϑ is *weakly optimal* if

$$\forall \tilde{\vartheta} \in \Theta \; \exists z \in Z \; F_z(\tilde{\vartheta}) \geq F_z(\vartheta)$$

Definition

The *global depth* of a model ϑ is defined as

$$d_G(\vartheta) = \min_{\tilde{\vartheta}} |\{ z \in Z \mid F_z(\tilde{\vartheta}) \geq F(\vartheta) \}|$$

Depth of fit

- Statistical model with *parameters* $\vartheta = (\vartheta_1 \ldots \vartheta_p) \in \Theta$
- Datapoints Z
- *Criterial Functions* $F_z : \Theta \to [0, \infty)$, $z \in Z$

Definition

Model ϑ is *weakly optimal* if

$$\forall \tilde{\vartheta} \in \Theta \; \exists z \in Z \; F_z(\tilde{\vartheta}) \geq F_z(\vartheta)$$

Definition

The *global depth* of a model ϑ is defined as

$$d_G(\vartheta) = \min_{\tilde{\vartheta}} |\{ z \in Z \mid F_z(\tilde{\vartheta}) \geq F(\vartheta) \}|$$

Depth of fit

- Statistical model with *parameters* $\vartheta = (\vartheta_1 \dots \vartheta_p) \in \Theta$
- Datapoints Z
- *Criterial Functions* $F_z : \Theta \to [0, \infty)$, $z \in Z$

Definition

Model ϑ is *weakly optimal* if

$$\forall \tilde{\vartheta} \in \Theta \; \exists z \in Z \; F_z(\tilde{\vartheta}) \geq F_z(\vartheta)$$

Definition

The *global depth* of a model ϑ is defined as

$$d_G(\vartheta) = \min_{\tilde{\vartheta}} |\{ z \in Z \mid F_z(\tilde{\vartheta}) \geq F(\vartheta) \}|$$

Linearization

Definition

For F_z differentiable, define the *tangent depth* of ϑ as

$$d_T(\vartheta) = \min_{u \neq 0} |\{ z \mid \langle u, \nabla F_z(\vartheta) \rangle \geq 0 \}|$$

Theorem (Mizera 2002)

If the F_z are differentiable and convex, and $\Theta \subset \mathbb{R}^p$ is open and convex, then for any model $\vartheta \in \Theta$

$$d_G(\vartheta) = d_T(\vartheta)$$

Linearization

Definition

For F_z differentiable, define the *tangent depth* of ϑ as

$$d_T(\vartheta) = \min_{u \neq 0} |\{ z \mid \langle u, \nabla F_z(\vartheta) \rangle \geq 0 \}|$$

Theorem (Mizera 2002)

If the F_z are differentiable and convex, and $\Theta \subset \mathbb{R}^p$ is open and convex, then for any model $\vartheta \in \Theta$

$$d_G(\vartheta) = d_T(\vartheta)$$

Example: Two Factor ANOVA

- Two different *experimental factors* with *levels* in $N = \{ 1 \ldots n \}$ and $M = \{ 1 \ldots m \}$.
- For each experimental setting (i, j) we have r data points $z_{i,j,1} \ldots z_{i,j,r}$ measuring outcomes.

	Fertilizer	
soil	1	2
1	2	1
2	5	5

For simplicity, here $r = 1$

- The subset $\{ z_{i,j,k} \mid k = 1 \ldots r \}$ corresponding to an experimental scenario is fit to some linear function $f(\vartheta) = \mu_i + \nu_j$.

Example: Two Factor ANOVA

- Two different *experimental factors* with *levels* in $N = \{1 \dots n\}$ and $M = \{1 \dots m\}$.
- For each experimental setting (i, j) we have r data points $z_{i,j,1} \dots z_{i,j,r}$ measuring outcomes.

	Fertilizer	
soil	1	2
1	2	1
2	5	5

For simplicity, here $r = 1$

- The subset $\{ z_{i,j,k} \mid k = 1 \dots r \}$ corresponding to an experimental scenario is fit to some linear function $f(\vartheta) = \mu_i + \nu_j$.

Example: Two Factor ANOVA

- Two different *experimental factors* with *levels* in $N = \{1 \ldots n\}$ and $M = \{1 \ldots m\}$.
- For each experimental setting (i, j) we have r data points $z_{i,j,1} \ldots z_{i,j,r}$ measuring outcomes.

	Fertilizer	
soil	1	2
1	2	1
2	5	5

For simplicity, here $r = 1$

- The subset $\{ z_{i,j,k} \mid k = 1 \ldots r \}$ corresponding to an experimental scenario is fit to some linear function $f(\vartheta) = \mu_i + \nu_j$.

ANOVA example continued: Criterial Functions

soil	Fertilizer $\nu_1 = 1$	$\nu_2 = 2$
$\mu_1 = 1$	2	1
$\mu_2 = 2$	5	5

- Parameter vector $\vartheta = (\mu_1 \ldots \mu_n, \nu_1 \ldots \nu_m)$.
- Criterial functions

$$F_{i,j,k}(\vartheta) = \frac{(z_{i,j,k} - (\mu_i + \nu_j))^2}{2}$$

- $\nabla F_{i,j,k}(\vartheta) = -(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$

ANOVA example continued: scaled gradients

Scaling gradients

Recall $\nabla F_{i,j,k}(\vartheta) = -(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$.
For purposes of computing depth, we may consider

$$G_{i,j,k}(\vartheta) = -\operatorname{sign}(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$$

soil	Fertilizer $\nu_1 = 1$	$\nu_2 =$
$\mu_1 = 1$	2	1
$\mu_2 =$	5	5

$G_{i,j}(1,2,1,2)$

i \ j	1	2
1	$(0,0,0,0)$	$(1,0,0,1)$
2		

$\text{depth}_Z(0) =$

ANOVA example continued: scaled gradients

Scaling gradients

Recall $\nabla F_{i,j,k}(\vartheta) = -(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$.
For purposes of computing depth, we may consider

$$G_{i,j,k}(\vartheta) = -\operatorname{sign}(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$$

soil	Fertilizer $\nu_1 = 1$	$\nu_2 = 2$
$\mu_1 = 1$	2	1
$\mu_2 = 2$	5	5

$G_{i,j}(1,2,1,2)$

i	j: 1	2
1	$(0,0,0,0)$	$(1,0,0,1)$
2	$-(0,1,1,0)$	$-(0,1,0,1)$

$$\text{depth}_Z(0) = 1$$

ANOVA example continued: scaled gradients

Scaling gradients

Recall $\nabla F_{i,j,k}(\vartheta) = -(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$.
For purposes of computing depth, we may consider

$$G_{i,j,k}(\vartheta) = -\operatorname{sign}(z_{i,j,k} - \mu_i - \mu_j)(e_i, e_j)$$

soil	Fertilizer $\nu_1 = 1$	Fertilizer $\nu_2 = 1$
$\mu_1 = 1$	2	1
$\mu_2 = 4$	5	5

$G_{i,j}(1,4,1,1)$

i	j = 1	j = 2
1	$(0,0,0,0)$	$(1,0,0,1)$
2	$(0,0,0,0)$	$(0,0,0,0)$

$\text{depth}_Z(0) = 3$

Maximum feasible subsystem

- Maximum Feasible Subsystem

  Given Infeasible system $Ax < 0$

  Find A maximum subsystem of rows $\{ \langle a_i, x \rangle < 0 \mid i \in I \}$ that is feasible

- MaxFS APX-hard Amaldi and Kann, 1998
- MaxFS and halfspace depth are equivalent

$$\min_{u \neq 0} |\{ p \in S \mid \langle u, p \rangle \geq 0 \}| = |S| - \max_{u} |\{ p \in S \mid \langle u, p \rangle < 0 \}|$$

  Note condition $u \neq 0$ is unnecessary for *strict* MaxFS.

- Halfspace depth is APX-hard.

Maximum feasible subsystem

- Maximum Feasible Subsystem

  Given Infeasible system $Ax < 0$

  Find A maximum subsystem of rows $\{ \langle a_i, x \rangle < 0 \mid i \in I \}$ that is feasible

- MaxFS APX-hard Amaldi and Kann, 1998
- MaxFS and halfspace depth are equivalent

$$\min_{u \neq 0} |\{ p \in S \mid \langle u, p \rangle \geq 0 \}| = |S| - \max_{u} |\{ p \in S \mid \langle u, p \rangle < 0 \}|$$

  Note condition $u \neq 0$ is unnecessary for *strict* MaxFS.

- Halfspace depth is APX-hard.

Maximum feasible subsystem

- Maximum Feasible Subsystem

  Given: Infeasible system $Ax < 0$

  Find: A maximum subsystem of rows $\{\, \langle a_i, x \rangle < 0 \mid i \in I \,\}$ that is feasible

- MaxFS APX-hard Amaldi and Kann, 1998
- MaxFS and halfspace depth are equivalent

$$\min_{u \neq 0} |\{\, p \in S \mid \langle u, p \rangle \geq 0 \,\}| = |S| - \max_{u} |\{\, p \in S \mid \langle u, p \rangle < 0 \,\}|$$

  Note condition $u \neq 0$ is unnecessary for *strict* MaxFS.

- Halfspace depth is APX-hard.

Maximum feasible subsystem

- Maximum Feasible Subsystem

  Given: Infeasible system $Ax < 0$

  Find: A maximum subsystem of rows $\{ \langle a_i, x \rangle < 0 \mid i \in I \}$ that is feasible

- MaxFS APX-hard Amaldi and Kann, 1998
- MaxFS and halfspace depth are equivalent

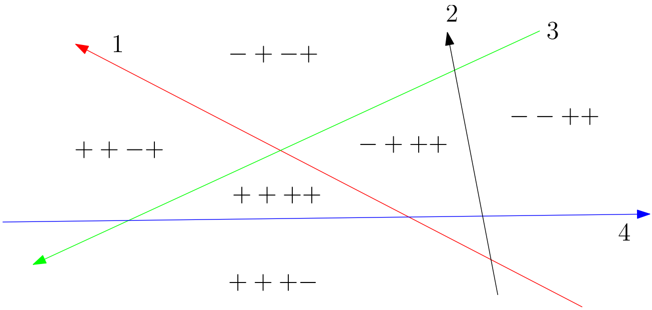
  $$\min_{u \neq 0} |\{ p \in S \mid \langle u, p \rangle \geq 0 \}| = |S| - \max_{u} |\{ p \in S \mid \langle u, p \rangle < 0 \}|$$

  Note condition $u \neq 0$ is unnecessary for *strict* MaxFS.

- Halfspace depth is APX-hard.

Traversing the dual arrangement

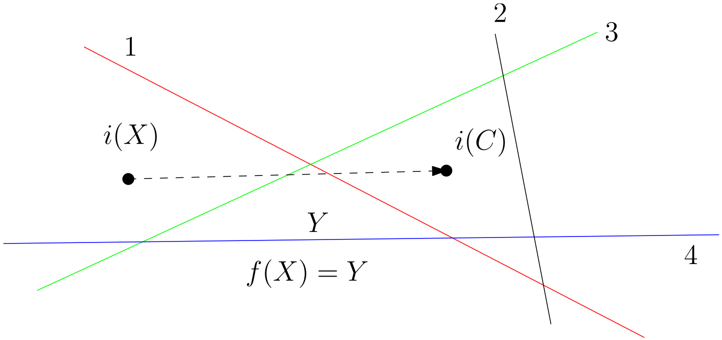
- $\mathrm{Adj}(X, j)$ is true iff negating sign j yields a cell. Test given polyhedron for interior. Solve via LP.



$\mathrm{Adj}(+ + ++, 1) =$ true $\mathrm{Adj}(+ + ++, 2) =$ false

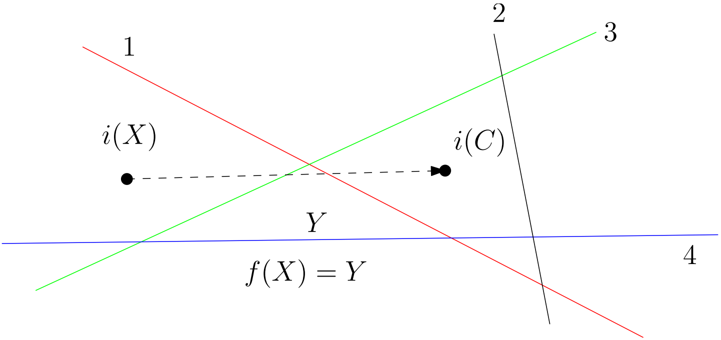
Moving towards the root

- Define a canonical interior point $i(X)$ for each cell. Same LP as before.
- Choose an arbitrary cell C.
- To find a closer cell to C "shoot a ray" from $i(X)$ to $i(C)$.



Moving towards the root

- Define a canonical interior point $i(X)$ for each cell. Same LP as before.
- Choose an arbitrary cell C.
- To find a closer cell to C "shoot a ray" from $i(X)$ to $i(C)$.



Reverse Search Summary

Theorem (FR04)

The halfspace depth of a point can be computed in $O(n \cdot \mathrm{LP}(n, d) \cdot (\#\ cells))$ *and* $O(nd)$ *space.*

- Optimizations include
  - Choosing a deep start cell
  - Pruning the search.
- Little information until enumeration terminates.

Reverse Search Summary

Theorem (FR04)

The halfspace depth of a point can be computed in $O(n \cdot \mathrm{LP}(n,d) \cdot (\#\ \mathit{cells}))$ and $O(nd)$ space.

- Optimizations include
  - Choosing a deep start cell
  - Pruning the search.
- Little information until enumeration terminates.

Reverse Search Summary

Theorem (FR04)

The halfspace depth of a point can be computed in $O(n \cdot \mathrm{LP}(n, d) \cdot (\#\ \textit{cells}))$ and $O(nd)$ space.

- Optimizations include
  - Choosing a deep start cell
  - Pruning the search.
- Little information until enumeration terminates.

Reverse Search Summary

Theorem (FR04)

The halfspace depth of a point can be computed in $O(n \cdot \mathrm{LP}(n, d) \cdot (\#\ \mathit{cells}))$ and $O(nd)$ space.

- Optimizations include
  - Choosing a deep start cell
  - Pruning the search.
- Little information until enumeration terminates.

Primal–Dual Algorithms

- Update at a every step an upper bound and a lower bound for the depth.
- Terminate when (if) bounds are equal
- To ensure termination, fall back on enumeration after a fixed time limit.
- Generally, answers improve with time.

Primal–Dual Algorithms

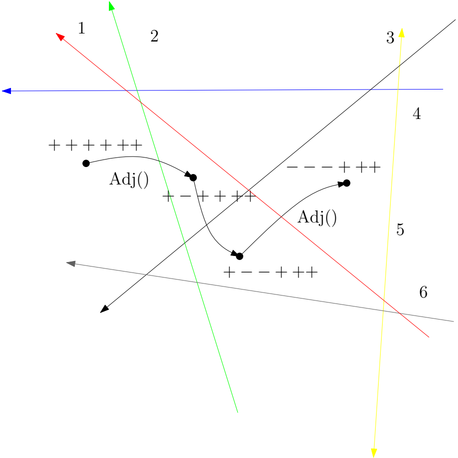
- Update at a every step an upper bound and a lower bound for the depth.
- Terminate when (if) bounds are equal
- To ensure termination, fall back on enumeration after a fixed time limit.
- Generally, answers improve with time.

Primal–Dual Algorithms

- Update at a every step an upper bound and a lower bound for the depth.
- Terminate when (if) bounds are equal
- To ensure termination, fall back on enumeration after a fixed time limit.
- Generally, answers improve with time.

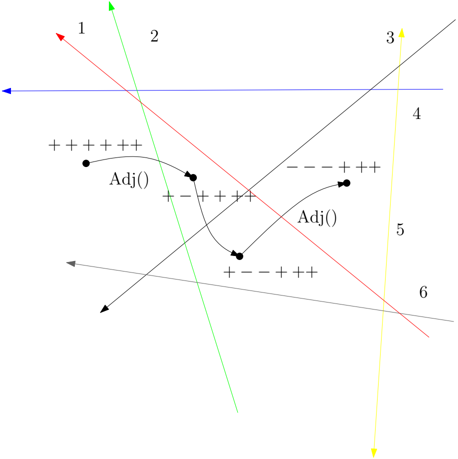
Upper Bounds via Random Walks

- Use Adj() oracle from enumeration algorithm
- Greedily try to reduce number of $+$ in σ until local minimum reached.
- Repeat several times choosing a random starting cell.



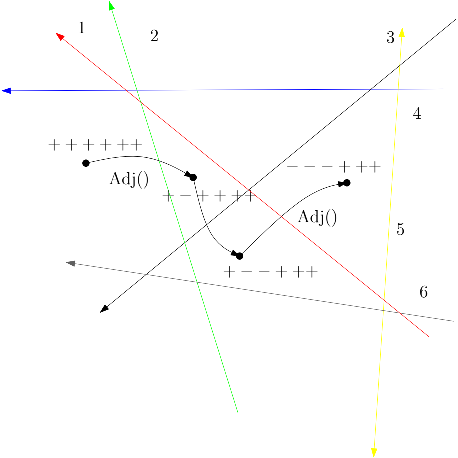
Upper Bounds via Random Walks

- Use Adj() oracle from enumeration algorithm
- Greedily try to reduce number of $+$ in σ until local minimum reached.
- Repeat several times choosing a random starting cell.



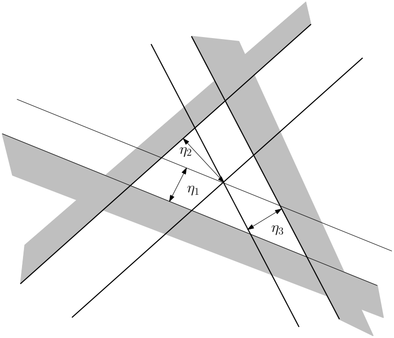
Upper Bounds via Random Walks

- Use Adj() oracle from enumeration algorithm
- Greedily try to reduce number of $+$ in σ until local minimum reached.
- Repeat several times choosing a random starting cell.



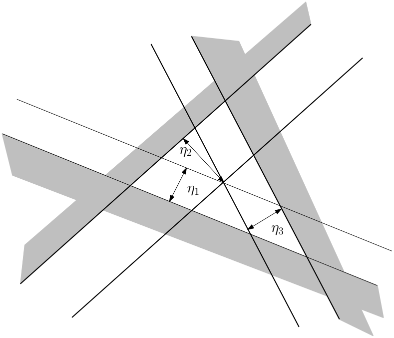
Upper Bounds Via Chinneck's Heuristic

- Elasticize: $a_i^T x < 0 \Rightarrow a_i^T x - \eta_i < 0, \eta \geq 0$
- Solve LP, $\min \mathrm{SINF} = \sum \eta_i$
- For each constraint j with $\eta_j > 0$, remove and resolve.
- Permanently remove the constraint that most improved SINF



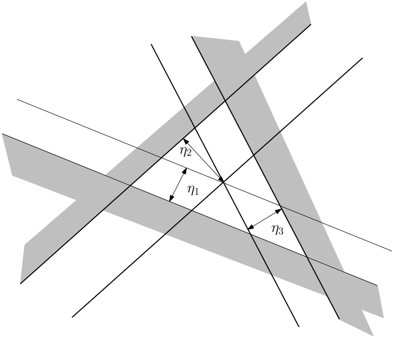
Upper Bounds Via Chinneck's Heuristic

- Elasticize: $a_i^T x < 0 \Rightarrow a_i^T x - \eta_i < 0, \eta \geq 0$
- Solve LP, $\min \mathrm{SINF} = \sum \eta_i$
- For each constraint j with $\eta_j > 0$, remove and resolve.
- Permanently remove the constraint that most improved SINF



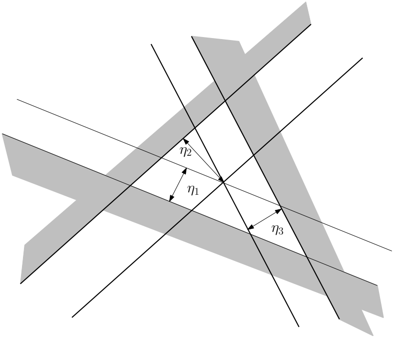
Upper Bounds Via Chinneck's Heuristic

- Elasticize: $a_i^T x < 0 \Rightarrow a_i^T x - \eta_i < 0, \eta \geq 0$
- Solve LP, $\min \mathrm{SINF} = \sum \eta_i$
- For each constraint j with $\eta_j > 0$, remove and resolve.
- Permanently remove the constraint that most improved SINF



Upper Bounds Via Chinneck's Heuristic

- Elasticize: $a_i^T x < 0 \Rightarrow a_i^T x - \eta_i < 0, \eta \geq 0$
- Solve LP, $\min \mathrm{SINF} = \sum \eta_i$
- For each constraint j with $\eta_j > 0$, remove and resolve.
- Permanently remove the constraint that most improved SINF

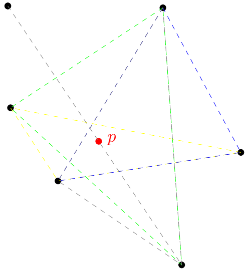


Lower Bounds via Minimal Dominating Sets

Definition

A *Minimal Dominating Set* (MDS) for $p \in \mathbb{R}^d$ with respect to $S \subset \mathbb{R}^d$ is $R \subseteq S$ such that

- $p \in \operatorname{conv} R$
- if $R' \subsetneq R$ then $p \notin \operatorname{conv} R'$.



Proposition

Let Δ be the set of all MDS's for p with respect to S. Let T be a minimum transversal (hitting set) of Δ.

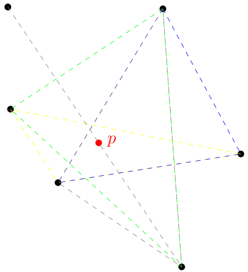
$$|T| = \operatorname{depth}(p)$$

Lower Bounds via Minimal Dominating Sets

Definition

A *Minimal Dominating Set* (MDS) for $p \in \mathbb{R}^d$ with respect to $S \subset \mathbb{R}^d$ is $R \subseteq S$ such that

- $p \in \operatorname{conv} R$
- if $R' \subsetneq R$ then $p \notin \operatorname{conv} R'$.

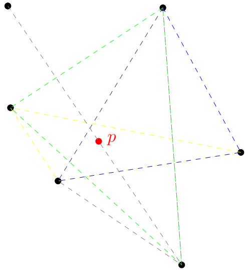


Proposition

Let Δ be the set of all MDS's for p with respect to S. Let T be a minimum transversal (hitting set) of Δ.

$$|T| = \operatorname{depth}(p)$$

Lower Bounds via Minimal Dominating Sets



Definition

A *Minimal Dominating Set* (MDS) for $p \in \mathbb{R}^d$ with respect to $S \subset \mathbb{R}^d$ is $R \subseteq S$ such that

- $p \in \operatorname{conv} R$
- if $R' \subsetneq R$ then $p \notin \operatorname{conv} R'$.

Proposition

Let Δ be the set of all MDS's for p with respect to S. Let T be a minimum transversal (hitting set) of Δ.

$$|T| = \operatorname{depth}(p)$$

Generating Missed MDSs (cuts)

Definition

Given a partial traversal T for the MDS's of p w.r.t. S, define $\bar{S} = S \setminus T$. Define the *auxiliary polytope* $Q(p, T)$ as λ satisfying:

$$\lambda \bar{S} = p$$

$$\sum_i \lambda_i = 1 \qquad \lambda_i \geq 0$$

- Each vertex (basic solution) of $Q(p, T)$ defines an MDS missed by T.
- A single cut can be found by LP
- k cuts can be found via reverse search (or other pivoting method).

Generating Missed MDSs (cuts)

Definition

Given a partial traversal T for the MDS's of p w.r.t. S, define $\bar{S} = S \setminus T$. Define the *auxiliary polytope* $Q(p, T)$ as λ satisfying:

$$\lambda \bar{S} = p$$

$$\sum_i \lambda_i = 1 \qquad \lambda_i \geq 0$$

- Each vertex (basic solution) of $Q(p, T)$ defines an MDS missed by T.
- A single cut can be found by LP
- k cuts can be found via reverse search (or other pivoting method).

Generating Missed MDSs (cuts)

Definition

Given a partial traversal T for the MDS's of p w.r.t. S, define $\bar{S} = S \setminus T$. Define the *auxiliary polytope* $Q(p, T)$ as λ satisfying:

$$\lambda \bar{S} = p$$

$$\sum_i \lambda_i = 1 \qquad \lambda_i \geq 0$$

- Each vertex (basic solution) of $Q(p, T)$ defines an MDS missed by T.
- A single cut can be found by LP
- k cuts can be found via reverse search (or other pivoting method).

Primal–Dual Algorithm

Implemented (BFR06) using ZRAM, cddlib, lrslib

1. Find candidate cell in the dual arrangement by upper bound heuristic
2. Find obstructions (i.e. MDS's) to the optimality of this cell
3. If none found, report optimal (we have solved the global minimum transversal problem).
4. Otherwise solve the resulting (partial) hitting set problem (or just find lower bound)
5. If bored, switch to enumeration.

Primal–Dual Algorithm

Implemented (BFR06) using ZRAM, cddlib, lrslib

1. Find candidate cell in the dual arrangement by upper bound heuristic
2. Find obstructions (i.e. MDS's) to the optimality of this cell
3. If none found, report optimal (we have solved the global minimum transversal problem).
4. Otherwise solve the resulting (partial) hitting set problem (or just find lower bound)
5. If bored, switch to enumeration.

Primal–Dual Algorithm

Implemented (BFR06) using ZRAM, cddlib, lrslib

1. Find candidate cell in the dual arrangement by upper bound heuristic
2. Find obstructions (i.e. MDS's) to the optimality of this cell
3. If none found, report optimal (we have solved the global minimum transversal problem).
4. Otherwise solve the resulting (partial) hitting set problem (or just find lower bound)
5. If bored, switch to enumeration.

Primal–Dual Algorithm

Implemented (BFR06) using ZRAM, cddlib, lrslib

1. Find candidate cell in the dual arrangement by upper bound heuristic
2. Find obstructions (i.e. MDS's) to the optimality of this cell
3. If none found, report optimal (we have solved the global minimum transversal problem).
4. Otherwise solve the resulting (partial) hitting set problem (or just find lower bound)
5. If bored, switch to enumeration.

Primal–Dual Algorithm

Implemented (BFR06) using ZRAM, cddlib, lrslib

1. Find candidate cell in the dual arrangement by upper bound heuristic
2. Find obstructions (i.e. MDS's) to the optimality of this cell
3. If none found, report optimal (we have solved the global minimum transversal problem).
4. Otherwise solve the resulting (partial) hitting set problem (or just find lower bound)
5. If bored, switch to enumeration.

Basic Infeasible Subsets

Definition

Let S be set of linear inequalities in ambient dimension d. A *basic infeasible subsystem* of S is a subset of at most $d+1$ inequalities that is infeasible.

Proposition

Let $Ax \geq b$ be an infeasible linear system. Any basic optimal solution to

$$\min \varepsilon$$

subject to

$$Ax + \varepsilon \geq b$$

defines a basic infeasible subsystem.

Basic Infeasible Subsets

Definition

Let S be set of linear inequalities in ambient dimension d. A *basic infeasible subsystem* of S is a subset of at most $d + 1$ inequalities that is infeasible.

Proposition

Let $Ax \geq b$ be an infeasible linear system. Any basic optimal solution to

$$\min \varepsilon$$

$$\textit{subject to}$$

$$Ax + \varepsilon \geq b$$

defines a basic infeasible subsystem.

Bounded depth exhaustive search

Algorithm $MFS(H : \text{halfspaces}, k : \text{integer})$

$B \leftarrow \text{BIS}(H)$
if $B = \emptyset$ then return true
if $k = 0$ then return false
for $h \in B$ do
  if $\text{MFS}(H \setminus h, k-1) =$ true then return true
endfor
return false

end

Theorem (BCILM06)

The halfspace depth of a point p with respect to a set S of n points in $\mathbb{R}^d$ can be computed in $O((d+1)^k LP(n, d-1))$ time, where k is the value of the output.

Bounded depth exhaustive search

Algorithm $MFS(H : \text{halfspaces}, k : \text{integer})$

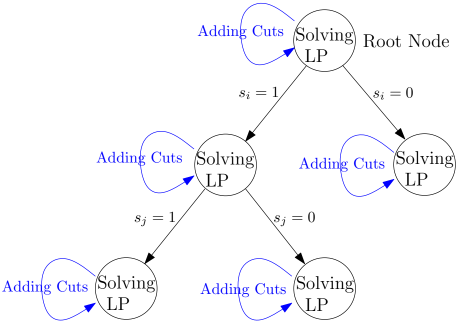
```
B ← BIS(H)
if B = ∅ then return true
if k = 0 then return false
for h ∈ B do
    if MFS(H \ h, k − 1) = true then return true
endfor
return false
```

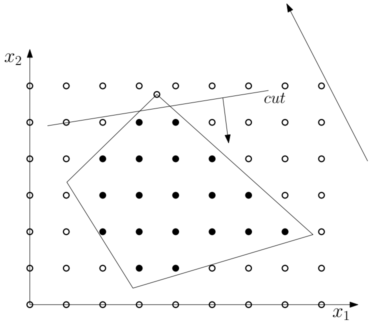
end

Theorem (BCILM06)

The halfspace depth of a point p with respect to a set S of n points in $\mathbb{R}^d$ can be computed in $O((d+1)^k LP(n, d-1))$ time, where k is the value of the output.

Branch and Cut





MIP formulation

Max Feasible Subsystem Problem

$$\max_x |\{ a_i \in A \mid \langle a_i, x \rangle < 0 \}|$$

Mixed Integer Program

$$\min \sum_i s_i$$

subj. to

$$\langle a_i, x \rangle - s_i M + \varepsilon \leq 0$$

MIP formulation

Max Feasible Subsystem Problem

$$\max_x |\{ a_i \in A \mid \langle a_i, x \rangle < 0 \}|$$

Mixed Integer Program

$$\min \sum_i s_i$$

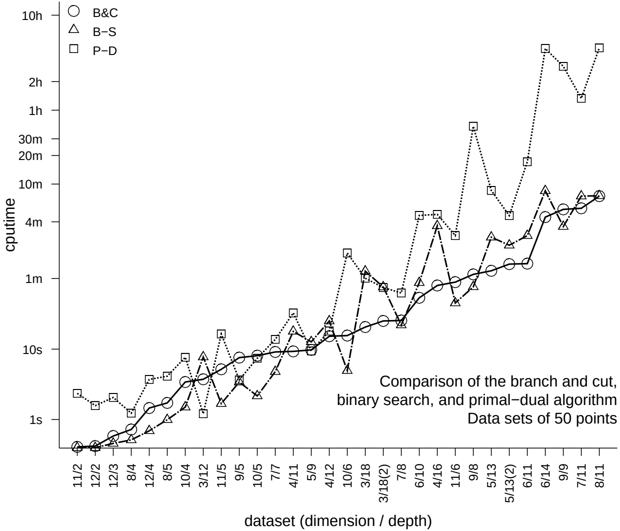
subj. to

$$\langle a_i, x \rangle - s_i M + \varepsilon \leq 0$$

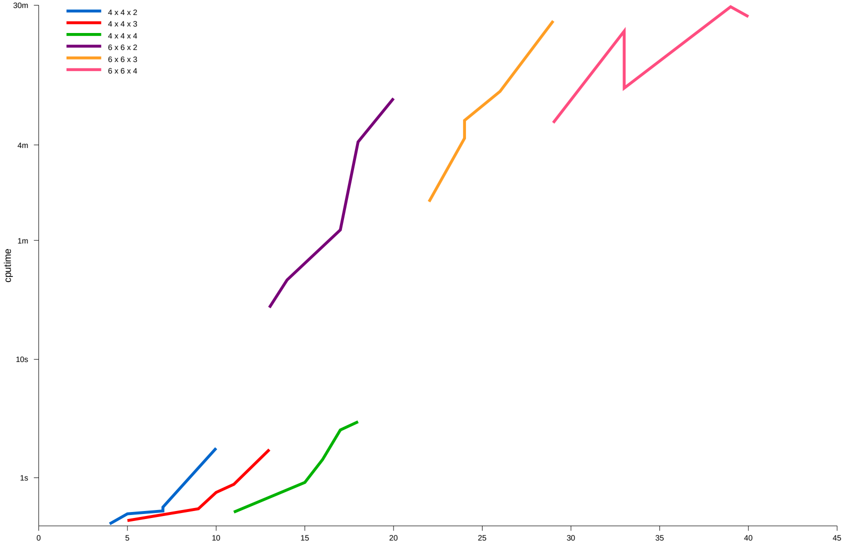
Branch and cut details

- Implementation by Dan Chen, using tools from COIN-OR.
- Chinneck's heuristic algorithm is used to find an initial upper bound
- MDS/BIS used as cutting planes.
- Binary-search version "eliminates" ε
- Various branching heuristics available.

Random Data


B&C
B−S
P−D
10h
2h
1h
30m
20m
10m
4m
1m
10s
1s
cputime
11/2
12/2
12/3
8/4
12/4
8/5
10/4
3/12
11/5
9/5
10/5
7/7
4/11
5/9
4/12
10/6
3/18
3/18(2)
7/8
6/10
4/16
11/6
9/8
5/13
5/13(2)
6/11
6/14
9/9
7/11
8/11
dataset (dimension / depth)
Comparison of the branch and cut,
binary search, and primal−dual algorithm
Data sets of 50 points

ANOVA Data



30m
4m
1m
10s
1s
cputime
0
5
10
15
20
25
30
35
40
45
4 x 4 x 2
4 x 4 x 3
4 x 4 x 4
6 x 6 x 2
6 x 6 x 3
6 x 6 x 4

Future work

Refinements

- More benchmark data
- Numerical issues
- Making B&C heuristics play nice together.
- Revisit primal–dual with better upper bounds
- Implement fixed parameter tractable algorithm, integrate with B&C

New directions

- Algorithms/Heuristics for centre
- Contours

Future work

Refinements

- More benchmark data
- Numerical issues
- Making B&C heuristics play nice together.
- Revisit primal–dual with better upper bounds
- Implement fixed parameter tractable algorithm, integrate with B&C

New directions

- Algorithms/Heuristics for centre
- Contours

Bibliography

David Bremner, Dan Chen, John Iacono, Stefan Langerman, and Pat Morin.
Output-sensitive algorithms for Tukey depth and related problems.
Submitted, September 2006.

David Bremner, Komei Fukuda, and Vera Rosta.
Primal dual algorithms for data depth.
In Reginia Y. Liu, Robert Serfling, and Diane L. Souvaine, editors, *Data Depth: Robust Multivariate Analysis, Computational Geometry, and Applications*, volume 72 of *AMS DIMACS Book Series*, pages 171–194. January 2006.

Dan Chen.
A branch and cut algorithm for the halfspace depth problem.
Master's thesis, UNB, 2007.

John Chinneck.
Fast heuristics for the maximum feasible subsystem problem.
INFORMS J. Computing, 13(3):210–223, 2001.

K. Fukuda and V. Rosta.
Exact parallel algorithms for the location depth and the maximum feasible subsystem problems.
In *Frontiers in global optimization*, volume 74 of *Nonconvex Optim. Appl.*, pages 123–133. Kluwer Acad. Publ., Boston, MA, 2004.

Ivan Mizera.
On depth and deep points: a calculus.
Ann. Statist., 30(6):1681–1736, 2002.