Greetings, Martin

Dyer and Volumes of Convex sets, ..

- # P-Hardness of computing the volume of a polyhedron Dyer and Frieze

Dyer and Volumes of Convex sets, ..

- # P-Hardness of computing the volume of a polyhedron Dyer and Frieze
- Random polynomial time algorithm Dyer, Frieze, K.

Dyer and Volumes of Convex sets, ..

- # P-Hardness of computing the volume of a polyhedron Dyer and Frieze
- Random polynomial time algorithm Dyer, Frieze, K.
- Improvement of the running time to $O^*(n^8)$ Dyer and Frieze

Dyer and Volumes of Convex sets, ..

- # P-Hardness of computing the volume of a polyhedron Dyer and Frieze
- Random polynomial time algorithm Dyer, Frieze, K.
- Improvement of the running time to $O^*(n^8)$ Dyer and Frieze
- (Exact) Number of Lattice Points in a fixed dimensional polyhedron - simplifying an algorithm of Barvinok - Dyer, K.

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:
  - Min $\quad F\left(c^{(1)}x, c^{(2)}x, \ldots, c^{(k)}x\right)$

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:
  - Min $\quad F\left(c^{(1)}x, c^{(2)}x, \ldots, c^{(k)}x\right)$
  - subject to $0 \leq x_i \leq 1$ and at most k of the x_i are fractional.

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:
  - Min $\quad F\left(c^{(1)}x, c^{(2)}x, \ldots, c^{(k)}x\right)$
  - subject to $0 \le x_i \le 1$ and at most k of the x_i are fractional.
- Further, time = time for gradient descent to solve the real relaxation Plus One Linear Program

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:
  - Min $F\left(c^{(1)}x, c^{(2)}x, \ldots, c^{(k)}x\right)$
  - subject to $0 \le x_i \le 1$ and at most k of the x_i are fractional.
- Further, time = time for gradient descent to solve the real relaxation Plus One Linear Program
- **Proof** Solve the real relaxation (no integrality) to get opt soln y.

Main Result

- $x = (x_1, x_2, \ldots, x_n)$ unknowns. $c^{(1)}, c^{(2)}, \ldots, c^{(k)}$ are k given n–vectors. $F : \mathbf{R}^k \to \mathbf{R}$ convex function.
- In poly time, we can find an (essentially) optimal solution to:
  - Min $\quad F\left(c^{(1)}x, c^{(2)}x, \ldots, c^{(k)}x\right)$
  - subject to $0 \le x_i \le 1$ and at most k of the x_i are fractional.
- Further, time = time for gradient descent to solve the real relaxation Plus One Linear Program
- **Proof** Solve the real relaxation (no integrality) to get opt soln y.
- Then, solve the LP: $c^{(1)}x = c^{(1)}y$; $\ldots$; $c^{(k)}x = c^{(k)}y$; $0 \le x_i \le 1$ to find a basic feasible solution which has at most k fractional x_i.

Quadratic Integer Programs

- Min $x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$, where, $k \in O(1)$ and f is convex.

Quadratic Integer Programs

- Min $x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$,
  where, $k \in O(1)$ and f is convex.
- If no f, semi-definite programme + Rounding can work. Eg. - Max Cut (where $A_{ij} = -$(edge weight)), ...

Quadratic Integer Programs

- Min $x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0,1\}$, where, $k \in O(1)$ and f is convex.
- If no f, semi-definite programme + Rounding can work. Eg. - Max Cut (where $A_{ij} = -$(edge weight)), ...
- Why f ? Can formulate: Max-Cut, Max-balanced cut, 2-means with given cluster sizes, 2-means with side information (i, j belong to same/different clusters).

Quadratic Integer Programs

- Min $x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$, where, $k \in O(1)$ and f is convex.
- If no f, semi-definite programme + Rounding can work. Eg. - Max Cut (where $A_{ij} = -$(edge weight)), ...
- Why f ? Can formulate: Max-Cut, Max-balanced cut, 2-means with given cluster sizes, 2-means with side information (i, j belong to same/different clusters).
- Seek: “Practical” algorithms with provable error guarantees. [Note: SDP squares number of variables.]

Example

- Given: n points $u_1, u_2, \ldots, u_n$ with $A_{ij} = |u_i - u_j|^2$. Partition into two sets of equal size minimizing the sum of squared distances of pairs of points in same part. Use convex penalty $M \left(\sum_i x_i - \frac{n}{2}\right)^2$ to balance the parts:

$$\text{Min } x^T A x + (\mathbf{1} - x)^T A (\mathbf{1} - x) + M \left(\sum_i x_i - \frac{n}{2} \right)^2 \text{ s.t. } x_i \in \{0, 1\}.$$

Example

- Given: n points $u_1, u_2, \ldots, u_n$ with $A_{ij} = |u_i - u_j|^2$. Partition into two sets of equal size minimizing the sum of squared distances of pairs of points in same part. Use convex penalty $M\left(\sum_i x_i - \frac{n}{2}\right)^2$ to balance the parts:

$$\text{Min } x^T A x + (\mathbf{1} - x)^T A (\mathbf{1} - x) + M\left(\sum_i x_i - \frac{n}{2}\right)^2 \text{ s.t. } x_i \in \{0, 1\}.$$

- Balanced 2-means clustering is a special case.

Example

- Given: n points $u_1, u_2, \ldots, u_n$ with $A_{ij} = |u_i - u_j|^2$. Partition into two sets of equal size minimizing the sum of squared distances of pairs of points in same part. Use convex penalty $M\left(\sum_i x_i - \frac{n}{2}\right)^2$ to balance the parts:

$$\text{Min } x^T A x + (\mathbf{1} - x)^T A (\mathbf{1} - x) + M\left(\sum_i x_i - \frac{n}{2}\right)^2 \text{ s.t. } x_i \in \{0, 1\}.$$

- Balanced 2-means clustering is a special case.
- Can similarly do 2-clustering or learning mixtures of 2 Gaussians with any given cluster weights.

Example

- Given: n points $u_1, u_2, \ldots, u_n$ with $A_{ij} = |u_i - u_j|^2$. Partition into two sets of equal size minimizing the sum of squared distances of pairs of points in same part. Use convex penalty $M\left(\sum_i x_i - \frac{n}{2}\right)^2$ to balance the parts:

$$\text{Min } x^T A x + (\mathbf{1} - x)^T A (\mathbf{1} - x) + M\left(\sum_i x_i - \frac{n}{2}\right)^2 \text{ s.t. } x_i \in \{0, 1\}.$$

- Balanced 2-means clustering is a special case.
- Can similarly do 2-clustering or learning mixtures of 2 Gaussians with any given cluster weights.
- 2-clustering with side constraints: Recent question in ML: 2-cluster with side info: (a limited number k of) i, j belong to same or different clusters. Constraints: $x_i = x_j$ or $x_i = 1 - x_j$. Eliminate x_i. Solve QIP. If solution has l fractional variables, whole program has at most $k + l$ fractional variables. If k, l are small, error in rounding the fractional variables (arbitrarily) is small....

Solution

- Min$x^T A x + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.

Solution

- Min$x^T A x + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.
- Wlg assume A is symmetric.

Solution

- $\text{Min} x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.
- Wlg assume A is symmetric.
- Simple Case: A has only non-negative eigenvalues and has $O(1)$ rank.

Solution

- Min$x^TAx + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.
- Wlg assume A is symmetric.
- Simple Case: A has only non-negative eigenvalues and has $O(1)$ rank.
- Solve the relaxation to real-valued x_i. Convex Program.

Solution

- $\text{Min} x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.
- Wlg assume A is symmetric.
- Simple Case: A has only non-negative eigenvalues and has $O(1)$ rank.
- Solve the relaxation to real-valued x_i. Convex Program.
- How do we round to an integer solution (with provable guarantees)?

Solution

- Min$x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0, 1\}$.
- Wlg assume A is symmetric.
- Simple Case: A has only non-negative eigenvalues and has $O(1)$ rank.
- Solve the relaxation to real-valued x_i. Convex Program.
- How do we round to an integer solution (with provable guarantees)?
- If $u_1, u_2, \ldots, u_r$ are the eigenvectors of A (r =rank(A)) and $u_1 \cdot x = \gamma_1 . u_2 \cdot x = \gamma_2 . u_r \cdot x = \gamma_r , v_1 \cdot x = \delta_1 , \ldots v_k \cdot x = \delta_k$ is real opt., then solve an auxiliary Linear Program: $u_1 \cdot x = \gamma_1 . u_2 \cdot x = \gamma_2 . u_r \cdot x = \gamma_r , v_1 \cdot x = \delta_1 , \ldots v_k \cdot x = \delta_k \; x_i \in [0, 1]$ to find a basic feasible solution.

Solution

- Min $x^T Ax + f(v_1 \cdot x, v_2 \cdot x, \ldots, v_k \cdot x)$ s.t. $x_i \in \{0,1\}$.
- Wlg assume A is symmetric.
- Simple Case: A has only non-negative eigenvalues and has $O(1)$ rank.
- Solve the relaxation to real-valued x_i. Convex Program.
- How do we round to an integer solution (with provable guarantees)?
- If $u_1, u_2, \ldots, u_r$ are the eigenvectors of A (r =rank(A)) and $u_1 \cdot x = \gamma_1 . u_2 \cdot x = \gamma_2 . u_r \cdot x = \gamma_r , v_1 \cdot x = \delta_1 , \ldots v_k \cdot x = \delta_k$ is real opt., then solve an auxiliary Linear Program: $u_1 \cdot x = \gamma_1 . u_2 \cdot x = \gamma_2 . u_r \cdot x = \gamma_r , v_1 \cdot x = \delta_1 , \ldots v_k \cdot x = \delta_k \; x_i \in [0,1]$ to find a basic feasible solution.
- It has at most $r + k \in O(1)$ fractional variables. Round them arbitrarily....

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c^{(1)}} \cdot \mathbf{x}, \mathbf{c^{(2)}} \cdot \mathbf{x}, \ldots, \mathbf{c^{(s)}} \cdot \mathbf{x}\right)$ subject to $x_i \in \{0, 1\}$.

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c}^{(1)} \cdot \mathbf{x}, \mathbf{c}^{(2)} \cdot \mathbf{x}, \ldots, \mathbf{c}^{(s)} \cdot \mathbf{x}\right)$ subject to $x_i \in \{0, 1\}$.
- Suppose $\lambda_1(A), \lambda_2(A), \ldots$ are the eigenvalues of A in non-increasing order of absolute value

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c}^{(\mathbf{1})}\cdot\mathbf{x}, \mathbf{c}^{(\mathbf{2})}\cdot\mathbf{x}, \ldots, \mathbf{c}^{(\mathbf{s})}\cdot\mathbf{x}\right)$ subject to $x_i \in \{0, 1\}$.
- Suppose $\lambda_1(A), \lambda_2(A), \ldots$ are the eigenvalues of A in non-increasing order of absolute value
- Assume $\lambda_t(A) \geq -M$ for some $M > 0$ for all t.

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c}^{(1)}\cdot\mathbf{x}, \mathbf{c}^{(2)}\cdot\mathbf{x}, \ldots, \mathbf{c}^{(s)}\cdot\mathbf{x}\right)$ subject to $x_i \in \{0,1\}$.
- Suppose $\lambda_1(A), \lambda_2(A), \ldots$ are the eigenvalues of A in non-increasing order of absolute value
- Assume $\lambda_t(A) \geq -M$ for some $M > 0$ for all t.
- Let r be an arbitrary positive integer. We can solve **QIP** to additive error at most
  $|\lambda_1(A)|(\sqrt{n(r+s)} + r^2 + rs) + |\mathbf{c}|\sqrt{r+s} + n(M + |\lambda_{r+1}(A)|)$.
  in time $O(n^{5/2} + n^2 r)$.

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c^{(1)}} \cdot \mathbf{x}, \mathbf{c^{(2)}} \cdot \mathbf{x}, \ldots, \mathbf{c^{(s)}} \cdot \mathbf{x}\right)$ subject to $x_i \in \{0,1\}$.
- Suppose $\lambda_1(A), \lambda_2(A), \ldots$ are the eigenvalues of A in non-increasing order of absolute value
- Assume $\lambda_t(A) \geq -M$ for some $M > 0$ for all t.
- Let r be an arbitrary positive integer. We can solve **QIP** to additive error at most
  $|\lambda_1(A)|(\sqrt{n(r+s)} + r^2 + rs) + |\mathbf{c}|\sqrt{r+s} + n(M + |\lambda_{r+1}(A)|)$.
  in time $O(n^{5/2} + n^2 r)$.
- The complexity bottleneck is solving a Linear Program in n variables with $O(r)$ constraints which takes time $O(n^{5/2})$ using a recent algorithm of Yin-tat Lee and Adam Sidford

The Result

- **QIP** Min$\mathbf{x}^T A\mathbf{x} + f\left(\mathbf{c^{(1)}}\cdot\mathbf{x}, \mathbf{c^{(2)}}\cdot\mathbf{x}, \ldots, \mathbf{c^{(s)}}\cdot\mathbf{x}\right)$ subject to $x_i \in \{0,1\}$.
- Suppose $\lambda_1(A), \lambda_2(A), \ldots$ are the eigenvalues of A in non-increasing order of absolute value
- Assume $\lambda_t(A) \geq -M$ for some $M > 0$ for all t.
- Let r be an arbitrary positive integer. We can solve **QIP** to additive error at most
  $|\lambda_1(A)|(\sqrt{n(r+s)} + r^2 + rs) + |\mathbf{c}|\sqrt{r+s} + n(M + |\lambda_{r+1}(A)|)$.
  in time $O(n^{5/2} + n^2 r)$.
- The complexity bottleneck is solving a Linear Program in n variables with $O(r)$ constraints which takes time $O(n^{5/2})$ using a recent algorithm of Yin-tat Lee and Adam Sidford
- Often worst term nM. Max Cut case: n (Largest Pos eigenvalus of edge weight matrix).