Selected Nonlinear Optimization Applications

Yinyu Ye
Department of Management Science and Engineering
Stanford University
Stanford, CA 94305, U.S.A.

http://www.stanford.edu/~yyye

Chapters 7.2, 11.4, and Wikipedia

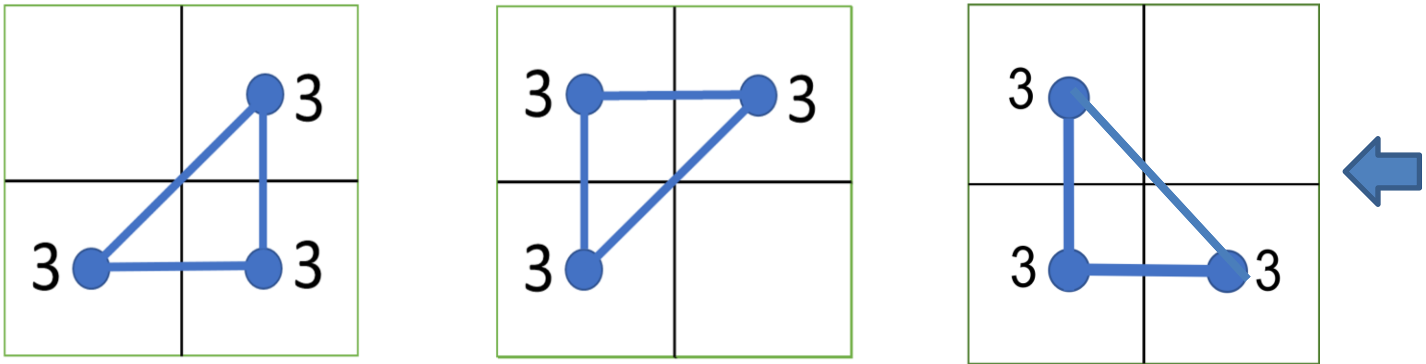
The Linear Regression: Least Squares Model

For the Business-or-Personal problem, we now minimize the sum of the squared errors (between predicted personal remittances and actual personal remittances).

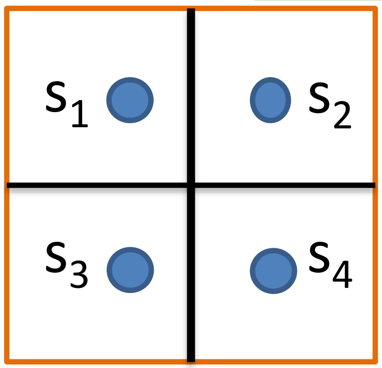
$$\text{Min} \quad \sum_i \left(\sum_j a_{ij} x_j - b_i \right)^2$$
$$\text{s.t.} \quad 0 \leq x_j \leq 1, \forall j.$$

- Let x_j be such a probability that a transaction is personal for industry code j
- $a_{i,j}$ – transaction amount for account i and industry code j
- b_i – amount paid by personal remit for account i
- $\sum_j a_{i,j} x_j$ – the expected personal expenses for account i
- We'd like to choose x_j such that $\sum_j a_{i,j} x_j$ matches b_i for ALL i
- This model is called Quadratic Optimization
- Convex? 1st Order Optimality Conditions? Sufficient?

The Wasserstein Barycenter Optimization



Three possible demand distribution scenario of 4 cities



Constraints:

$s_1+s_2+s_3+s_4=9$

$(s_1,s_2,s_3,s_4)>=0$

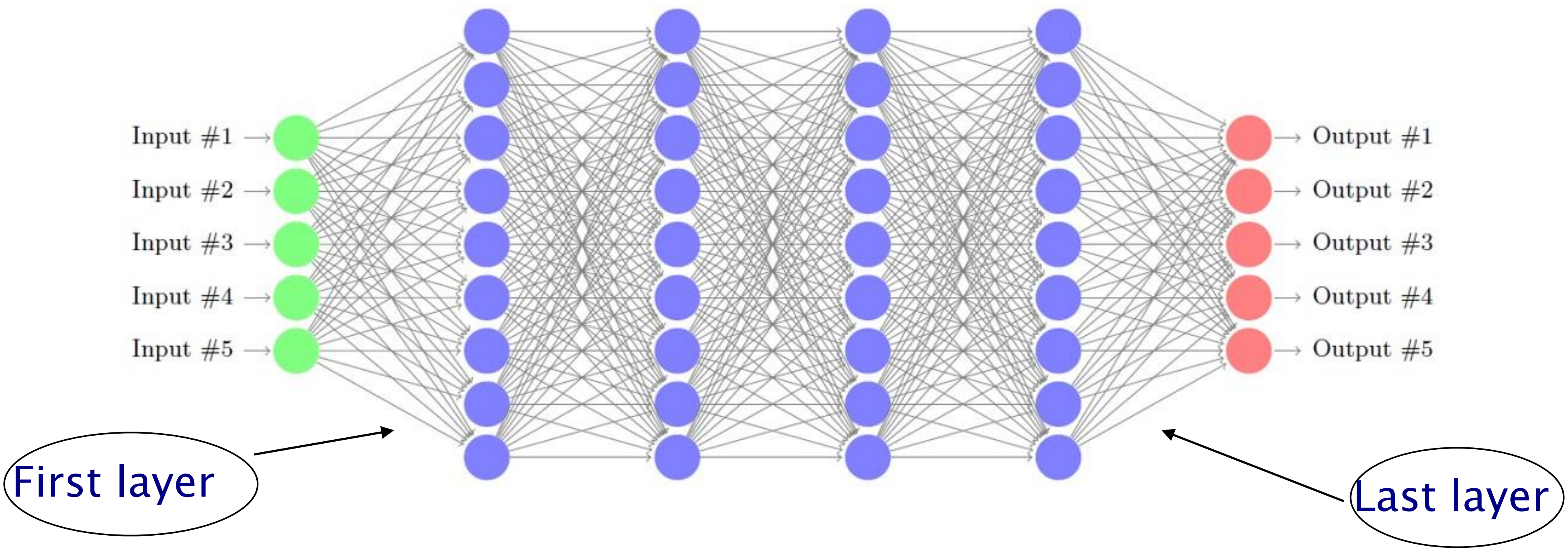
$$C=\begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 2 & 1 & 1 & 0 \end{pmatrix}$$

$$\begin{aligned} \text{minimize} \quad & WD_l(\boldsymbol{s}) + WD_r(\boldsymbol{s}) + WD_r(\boldsymbol{s}) \\ \text{s.t.} \quad & s_1+s_2+s_3+s_4 = 9, \;\; \boldsymbol{s} \geq \boldsymbol{0} \end{aligned}$$

Wasserstein-Distance function, WD($\boldsymbol{s}$), is an implicit nonlinear but convex function defined by the minimum value of a transportation minimization problem from supply inventory distribution $\boldsymbol{s}$.

This is a linearly constrained convex nonlinear optimization problem.

Nonlinear Optimization in Deep Learning



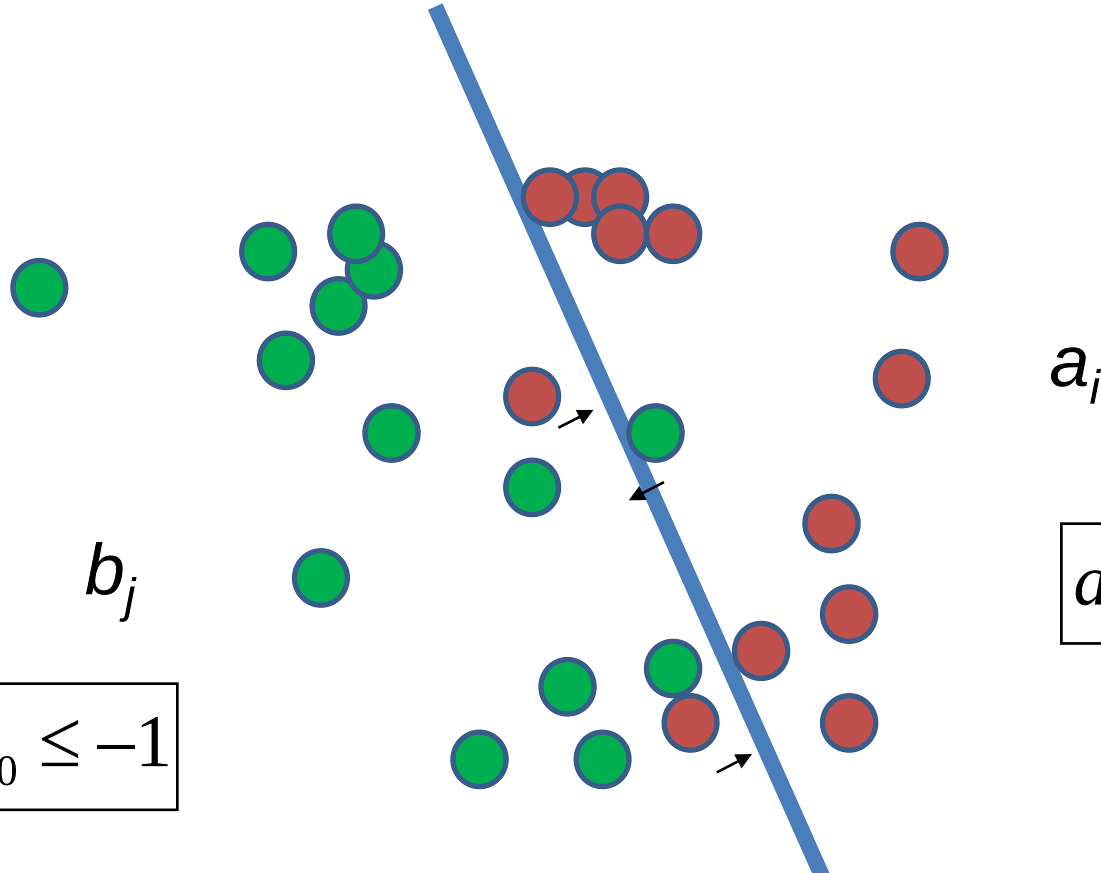
$$minimize \quad L(ReLu(a_i(w_{i,j})))$$

where $w_{i,j}$ is the weigh variable at laye i and edge j,

$(a_i(w_{i,j}))$ is a linear function of weights

ReLu(·) is the function max{· , 0} , called Rectified Linear Unit function (a type of neural behavior) in Deep Learning. The problem is typically Nonlinear and Nonconvex.

SVM with Quadratic Optimization I



$$a_i^T x + x_0 \geq 1$$

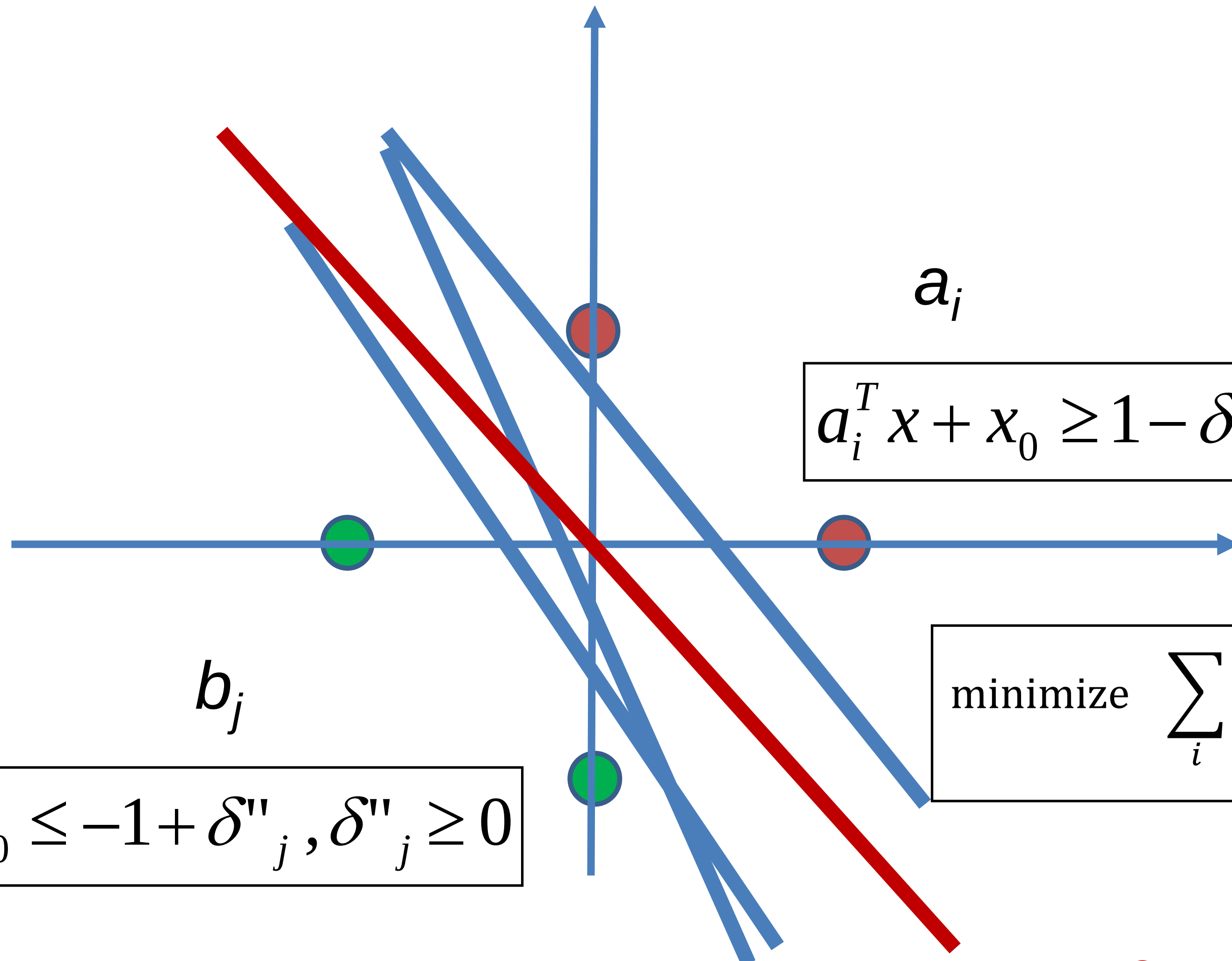
$$a_i^T x + x_0 \geq 1 - \delta'_i, \delta'_i \geq 0$$

$$b_j^T x + x_0 \leq -1$$

$$b_j^T x + x_0 \leq -1 + \delta''_j, \delta''_j \geq 0$$

$$\text{minimize} \quad \left(\sum_i \delta'_i + \sum_j \delta''_j \right) + \mu ||x||_2^2$$

SVM with Quadratic Optimization II



$$a_i^T x + x_0 \geq 1 - \delta'_i \,, \delta'_i \geq 0$$

$$\text{minimize} \quad \sum_i \delta'_i + \sum_j \delta''_j$$

$$b_j^T x + x_0 \leq -1 + \delta''_j \,, \delta''_j \geq 0$$

$$\text{minimize} \left(\sum_i \delta'_i + \sum_j \delta''_j \right) + 0.1 ||x||_2^2$$

Convex?
1st Order Optimality Conditions?
Sufficient?

SVM vs Logistic Regression: Likelihood Probability

Given message $\mathbf{a}_i$, according to the logistic model, the probability that it's a SPAM is represented by

$$\frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}.$$

Thus, for the training data, we like to determine x_0 and $\mathbf{x}$ from a set of training data (some spam some not) such that

$$\frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)} \sim \begin{cases} 1, & \text{spam} \\ 0, & \text{not.} \end{cases}$$

The probability to give a "right answer" for all training messages is

$$\left[\prod_{i:spam} \frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}\right]\left[\prod_{i:not}\left[1 - \frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}\right]\right]$$

Logistic Regression: Max-Likelihood Probability

Thus, we like to determine x_0 and $\mathbf{x}$ to maximize

$$\left[\prod_{i:spam} \frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}\right]\left[\prod_{i:not}\left[1 - \frac{\exp(\mathbf{a}_i^T\mathbf{x} + x_0)}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}\right]\right]$$

$$\Updownarrow$$

$$\left[\prod_{i:spam} \frac{1}{1 + \exp(-\mathbf{a}_i^T\mathbf{x} - x_0)}\right]\left[\prod_{i:not} \frac{1}{1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)}\right]$$

which is equivalent to maximize the log-likelihood probability

$$-\sum_{i:spam} \log\left(1 + \exp(-\mathbf{a}_i^T\mathbf{x} - x_0)\right) - \sum_{i:not} \log\left(1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)\right).$$

Or to minimize the convex (!) log-logistic-loss (with possible L_2 regularization term $\|\mathbf{x}\|^2$ added into the objective)

$$\sum_{i:spam} \log\left(1 + \exp(-\mathbf{a}_i^T\mathbf{x} - x_0)\right) + \sum_{i:not} \log\left(1 + \exp(\mathbf{a}_i^T\mathbf{x} + x_0)\right).$$

Convex?
1st Order Optimality Conditions?
Sufficient?

Convexity of log-exponential-sum

$$f(x_1, x_2) = \log(e^{x_1} + e^{x_2})$$

$$\nabla f(x_1, x_2) = \begin{pmatrix} \dfrac{e^{x_1}}{e^{x_1} + e^{x_2}} \\ \dfrac{e^{x_2}}{e^{x_1} + e^{x_2}} \end{pmatrix}$$

$$\nabla^2 f(x_1, x_2) = \frac{1}{(e^{x_1} + e^{x_2})^2} \begin{pmatrix} e^{x_1+x_2} & -e^{x_1+x_2} \\ -e^{x_1+x_2} & e^{x_1+x_2} \end{pmatrix}$$

The Hessian matrix is PSD everywhere so that LR is an Unconstrained Convex Program

Geometric Optimization and Convexification

$$\begin{array}{ll} \min & x \cdot y + y \cdot z + z \cdot x \\ \text{s.t.} & x \cdot y \cdot z = 1, \\ & (x, y, z) \geq 0 \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \min & e^{u+v} + e^{v+w} + e^{w+u} \\ \text{s.t.} & e^{u+v+w} = 1 \text{ or } u + v + w = 0. \end{array}$$

$$\text{Let } u = \ln(x), v = \ln(y), w = \ln(z), \text{ then}$$

This a Linearly Constrained (Convex) Optimization Program

Dynamic Optimal Pricing

p: price decision

$q(p) = \exp(a\log(p) + b)$: demand volume function

a: elasticity coefficient (< 0)

b: fixed and/or extenality coefficient

c: unit cost

Profit function: $(p\text{-}c)\cdot q(p) = (p\text{-}c)\cdot\exp(a\log(p) + b)$

Optimal price:

$$p^* = \begin{cases} \dfrac{ac}{a+1} & \text{if } a < -1 \\ c & \text{otherwise} \end{cases} : \text{optimal price to maximiza profit}$$

Note that the optimal price does not depend on b.

Use historical data and regression method to learn the demand function coefficients, and then compute the optimal price and fixed it for the remaining time periods.

Dynamic Pricing: update demand coefficients using new data and recalculate optimal price.

Online Combinatorial Auction I

Order fill	Price Limit π	Quantity Limit q	Argentina	Brazil	Italy	Germany	France
x1	0.75	10	1	1	1		
x2	0.35	5				1	
x3	0.40	10	1		1		1
x4	0.95	10	1	1	1	1	
x5	0.75	5		1		1	

$$\max_{\mathbf{x},w} \sum_{\mathrm{j}=1}^{\mathrm{n}} \pi_{\mathrm{j}} x_j - w$$

$$\text{s.t.} \quad \sum_{j=1}^{n} \mathbf{a}_j x_j - \mathbf{1}w \le \mathbf{0},$$

$$0 \le x_j \le q_j, \forall j$$

π_j: the jth bidding price

$\mathbf{a}_j$: the jth bidding vector;

q_j: the jth bidding share up limit;

Now Consider the online decisions

And the nonlinear optimization model

$$\max_{\{x,w,\mathbf{s}\}} \sum_{\mathrm{j}=1}^{\mathrm{n}} \pi_{\mathrm{j}} x_j - w + \sum_{i=1}^{m} u_i(s_i)$$

$$\text{s.t.} \quad \sum_{j=1}^{n} \mathbf{a}_j x_j - \mathbf{1}w + \mathbf{s} = \mathbf{0},$$

$$0 \le x_j \le q_j, \mathbf{s} \ge \mathbf{0}.$$

u(s): **Nonlinear Concave Value Function**

Logarithmic function with a positive weight μ

$$u(s) = \mu \ln(s)$$

Exponential function with a positive weight μ

$$u(s) = \mu(1 - \exp(-s/\mu))$$

Sequential Convex Programming Mechanism: OCA II

$$\begin{aligned} \max \quad & \sum_{j=1}^{n} \pi_j x_j - w \\ \text{s.t.} \quad & \sum_{j=1}^{n} \mathbf{a}_j x_j - \mathbf{1} w \le \mathbf{0}, \\ & 0 \le x_j \le q_j, \forall j \end{aligned}$$

Suppose the order fill decision x_j^* from $j=1$ to k have been made. Now the kth bidder arrives and we solve the following convex optimization problem

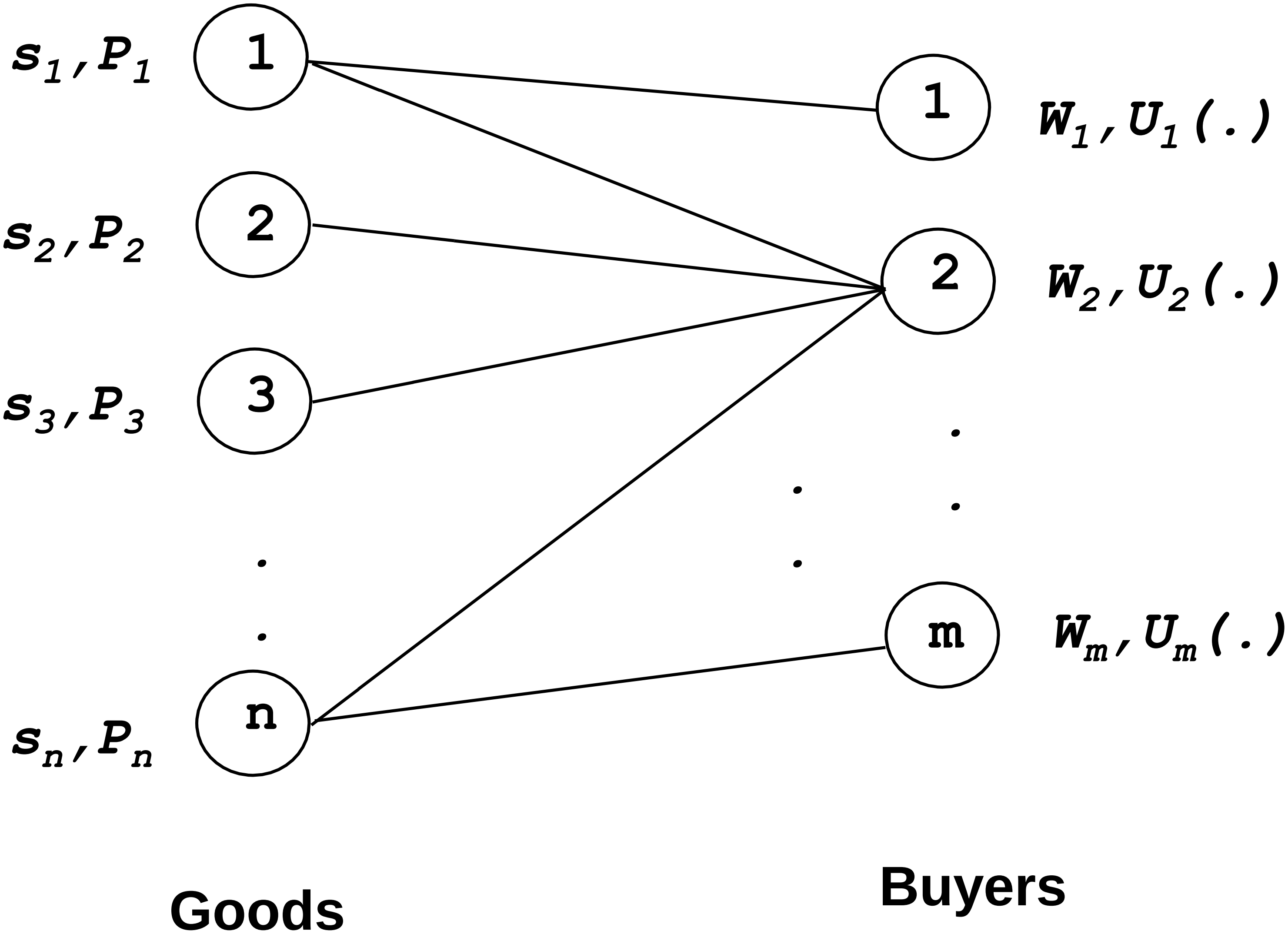
$$\begin{aligned} \max_{\{x,w,\mathbf{s}\}} \quad & \pi_k x - w + \sum_{i=1}^{m} u_i(s_i) \\ \text{s.t.} \quad & \mathbf{a}_k x - \mathbf{1} w + \mathbf{s} = -\sum_{j=1}^{k-1} a_j x_j^*, \\ & 0 \le x \le q_k, \quad \mathbf{s} \ge \mathbf{0}. \end{aligned}$$

Slack variables, or shares held by us.

Value functions to hold own shares

Outstanding shares already sold to bidders.

Fisher's Equilibrium Market Model



- **Buyers** have a money budget (w_i) to buy goods and maximize their individual utility functions $u(.)$;
- **Producers** have goods (s_j) to sell for money

- The equilibrium price is an assignment of prices (p_j) to goods so that every buyer could buy a maximal bundle of goods and also clear the market, meaning that all the money are spent and all goods are sold.

Each Buyer's Maximization and Equilibrium Price

Buyer $i = 1,\dots,m$ optimization problem for given prices p_j, $j = 1,\dots,n$.

$$\begin{aligned} \max_x \quad & \mathbf{u_i}^T \mathbf{x}_i := \textstyle\sum_j u_{ij} x_{ij} \\ \text{s.t.} \quad & \mathbf{p}^T \mathbf{x}_i := \textstyle\sum_j p_j x_{ij} \le w_i, \\ & x_{ij} \ge 0, \quad \forall \quad j, \end{aligned}$$

The equilibrium price vector $\boldsymbol{p}$ is the one to make

$$\sum_i x^*_{ij} = s_j$$

where $\boldsymbol{x}^*_i$ is an optimal solution vector of the above problem with given prices $\boldsymbol{p}$ for all $i=1,\dots,m$.

Example of Fisher's Model and Equilibrium Conditions

Buyer 1 and 2's optimization problems for given prices p_x, p_y on two goods x and y, each has one unit on the market.

$$\begin{array}{ll} \max & 2x_1 + y_1 \\ \text{s.t.} & p_x \cdot x_1 + p_y \cdot y_1 \le 5, \\ & x_1,\ y_1 \ge 0; \end{array}$$

$$\begin{array}{ll} \max & 3x_2 + y_2 \\ \text{s.t.} & p_x \cdot x_2 + p_y \cdot y_2 \le 8, \\ & x_2,\ y_2 \ge 0. \end{array}$$

$$\begin{array}{lll} p_x\lambda_1 - 2 \ge 0 & \wedge & x_1 \\ p_y\lambda_1 - 1 \ge 0 & \wedge & y_1 \\ p_x\lambda_2 - 3 \ge 0 & \wedge & x_2 \\ p_y\lambda_2 - 1 \ge 0 & \wedge & y_2 \end{array}$$

$$x_1 + x_2 = 1$$
$$y_1 + y_2 = 1$$

$$p_x = \frac{26}{3},\quad p_y = \frac{13}{3}$$

$$x_1 = \frac{6}{78},\ y_1 = 1,\ x_2 = \frac{72}{78},\ y_2 = 0$$

satisfy all these conditions so that the prices are equilibrium prices

Aggregate Social Optimization Problem

$$\begin{aligned} \max \quad & \textstyle\sum_i w_i \log(\mathbf{u}_i^T \mathbf{x}_i) \\ \text{s.t.} \quad & \textstyle\sum_i x_{ij} = s_j, \quad \forall \; j = 1,\dots,n \\ & x_{ij} \geq 0, \quad \forall \; i, j, \end{aligned}$$

Theorem *(Eisenberg and Gale 1959) The optimal Lagrange multiplier vector of the equality constraints of the social optimization problem is the equilibrium price vector.*

Aggregate Social Problem for the Example

$$\max \quad 5\log(2x_1 + y_1) + 8\log(3x_2 + y_2)$$

$$\text{s.t.} \quad x_1 + x_2 = 1,$$

$$y_1 + y_2 = 1,$$

$$(x_1, x_2, y_1, y_2) \geq 0.$$

$$\frac{10}{2x_1 + y_1} \leq p_x \wedge x_1 \qquad \frac{5}{2x_1 + y_1} \leq p_y \wedge y_1$$

$$\frac{24}{3x_2 + y_2} \leq p_x \wedge x_2 \qquad \frac{8}{3x_2 + y_2} \leq p_y \wedge y_2$$

The optimality conditions of the social optimization problem.

By assign

$5\lambda_1 = 2x_1 + y_1$

$8\lambda_2 = 3x_2 + y_2$

these conditions coincides The equilibrium conditions on Slide 16.

Sensor Network Localization I

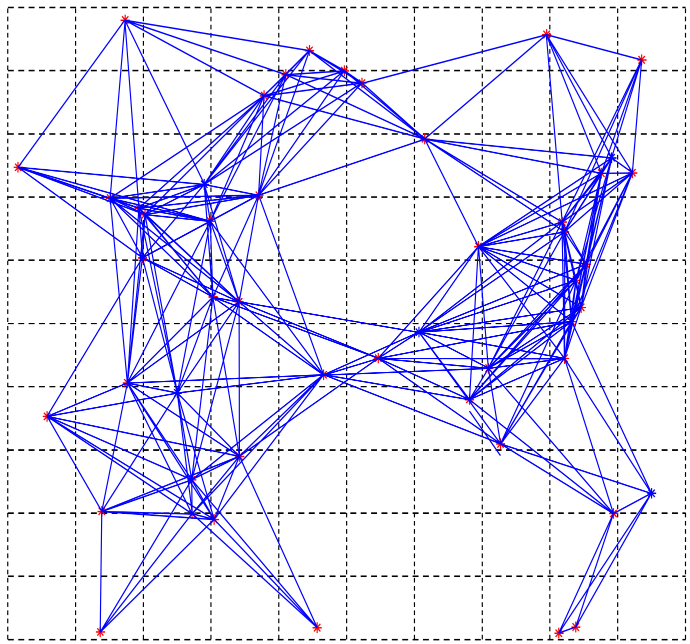
Given a graph $G = (V, E)$ and sets of partial distance measurements, say $\{d_{ij} : (i, j) \in E\}$, the goal is to compute a realization of G in the Euclidean space $\mathbf{R}^d$ for a given low dimension d, i.e.

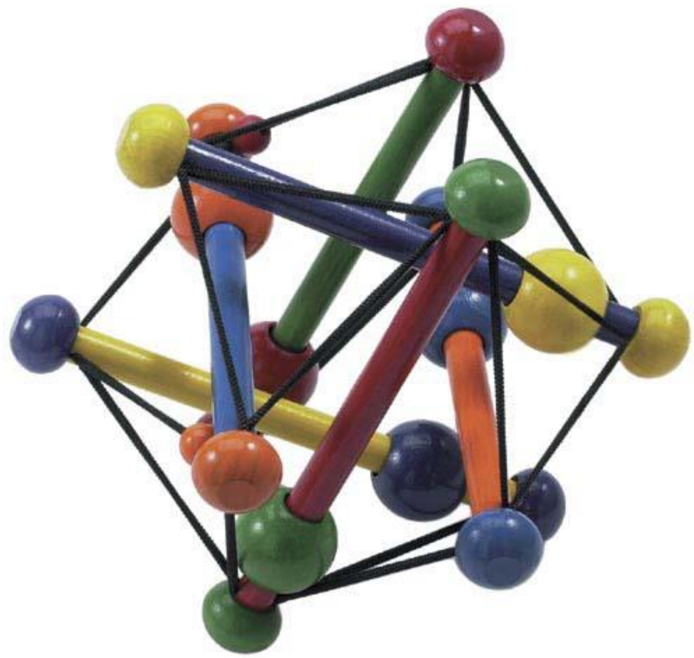
- to place the nodes/vertices of G in $\mathbf{R}^d$ such that
- the Euclidean distance between every pair of adjacent vertices $(i, j) \in E$ equals the measurements $d_{ij} \in E$.

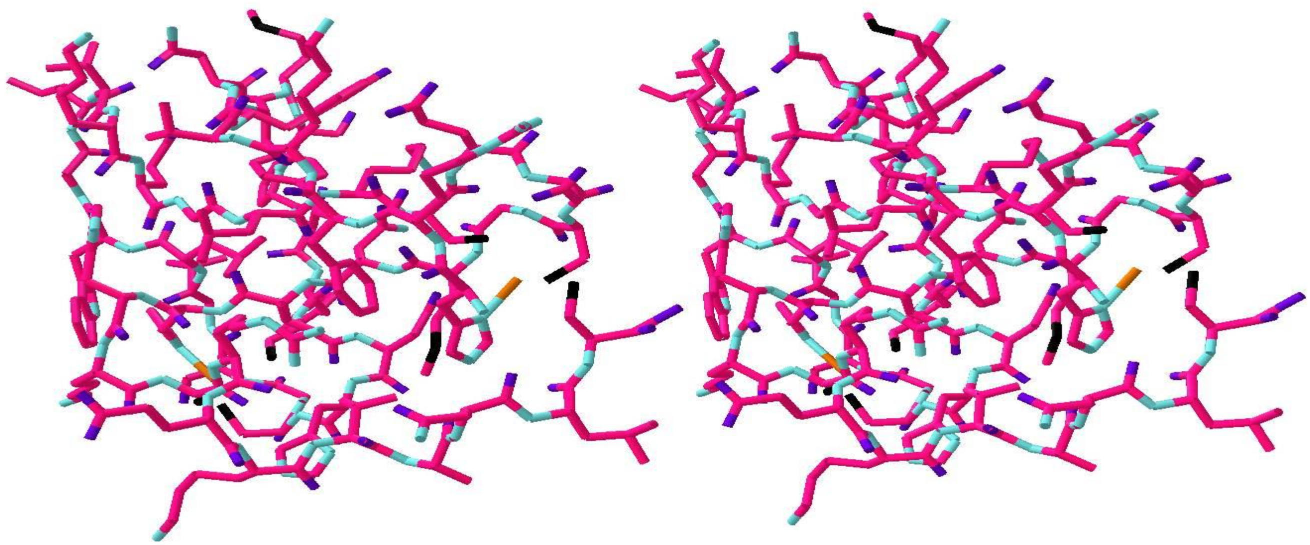
In general the localization may not be fixed since the configuration can rotate and translate. Thus, we assume that the positions of (d+1) sensors are known, and they called anchors.

This problem has wide applications …

Sensor Network Localization II







Sensor Network Localization III

Find d-dimensional points/vectors $\mathbf{x}_j$, $j=d+2,d+3,\ldots,n$, such that

$$
\begin{array}{l}
\| x_i - x_j \| = d_{ij} \quad \forall (i,j) \in E \\
\text{Anchors}: \quad x_i = a_i, \quad i = 1,2,\ldots,d+1
\end{array}
$$

This is a system of quadratic equations (after square both sides) and nonconvex, in contrast to a system of linear equations.

Does the system have a solution/localization of all $\mathbf{x}_j$'s? Is the solution/localization unique? Is there a certification for a solution to make it reliable or trustworthy? Is the system partially localizable with certification?

To get something tractable, we can consider optimization formulation or convex relaxation approaches.

SNL: Nonlinear Least Squares and Convex Relaxation

One can form a nonlinear least square minimization problem

$$\text{minimize} \quad \sum_{(ij)\text{in } E} \left(\|x_i - x_j\|^2 - (d_{ij})^2 \right)^2$$

$$\text{s.t.} \qquad x_i = a_i, \text{ for } i=1,\ldots,d+1$$

This remains nonconvex quartic polynomial minimization (with many local minimums)

We would use various algorithms to tackle the problem, including some convex relaxation approaches, which would be discussed later in the class.

$$\begin{aligned} &\text{minimize} \quad 0^T x: \\ &\text{s.t.} \qquad \|x_i - x_j\| \le (d_{ij}) \text{ for } (I,j) \text{ in } E \\ &\text{where} \quad x_i = a_i, \text{ for } i=1,\ldots,d+1 \end{aligned}$$

Convex?
1st Order
Optimality
Conditions?
Sufficient?