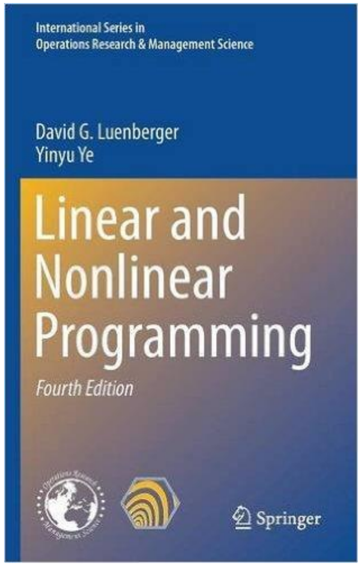
Introduction to Optimization

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

https://canvas.stanford.edu/courses/142617

Read Chapter 1.1, 1.2, 2.1, 2.2, Appendices A, B&D in Text-Book (hard copies would be available in the Book Store)

1^{st} Day Questions



- My CA team: Aldo, Luc, Chunlin, Mingxi
- Websites: Canvas
  https://canvas.stanford.edu/courses/142617
- 4 Homework assignments, 1 Take-Home Midterm, 1 team project
  - 40%*H + 30%*M+ 30%*P
  - No difference on taking 3 or 4 units
- No formula for cutoff between A/B etc.
- The more fun we all have, the more A's we will give out.
- Textbook: Linear and Nonlinear Programming (LY 4-5th edition, posted in Canvas)
- This is 111X/211X; if never had calculus and linear algebra classes, take 111/211
- The software use will help: Solvers in Matlab, R, Python or other public free software. It is mostly a "PAPER AND PENCIL" class!
- Form a "diversified" study group
- Selected Friday's problem sessions (will be taped)
- Students with OAE, extra two day for the exam

Mathematical Optimization Model

- Often consider the common quantitative model of data/decision/management science & engineering:

  - Maximize or Minimize $f(\boldsymbol{x})$
    for all $\boldsymbol{x} \in$ some set X

- Decision variables $\boldsymbol{x}$, Objective function $f(\boldsymbol{x})$, Constraint set X

- Applications in:
  - Applied Science, Engineering, Economics, Finance, Medicine, Statistics, Business
  - General Decision and Policy Making

- The famous Eighteenth Century Swiss mathematician and physicist Leonhard Euler (1707-1783) proclaimed that "…nothing at all takes place in the Universe in which some rule of maximum or minimum does not appear."

The Prototypical Optimization Problem

Max (or Max): $f(\mathbf{x})$

s.t. :

$$h_1(\mathbf{x}) = 0$$

$$\ldots$$

$$h_m(\mathbf{x}) = 0$$

$$g_1(\mathbf{x}) \leq 0$$

$$\ldots$$

$$g_r(\mathbf{x}) \leq 0$$

The Function could be:

x_1+2x_2, $x^2+2xy+2y^2$, $xln(x)+e^y$, $|x|+max\{x,y\}$, etc

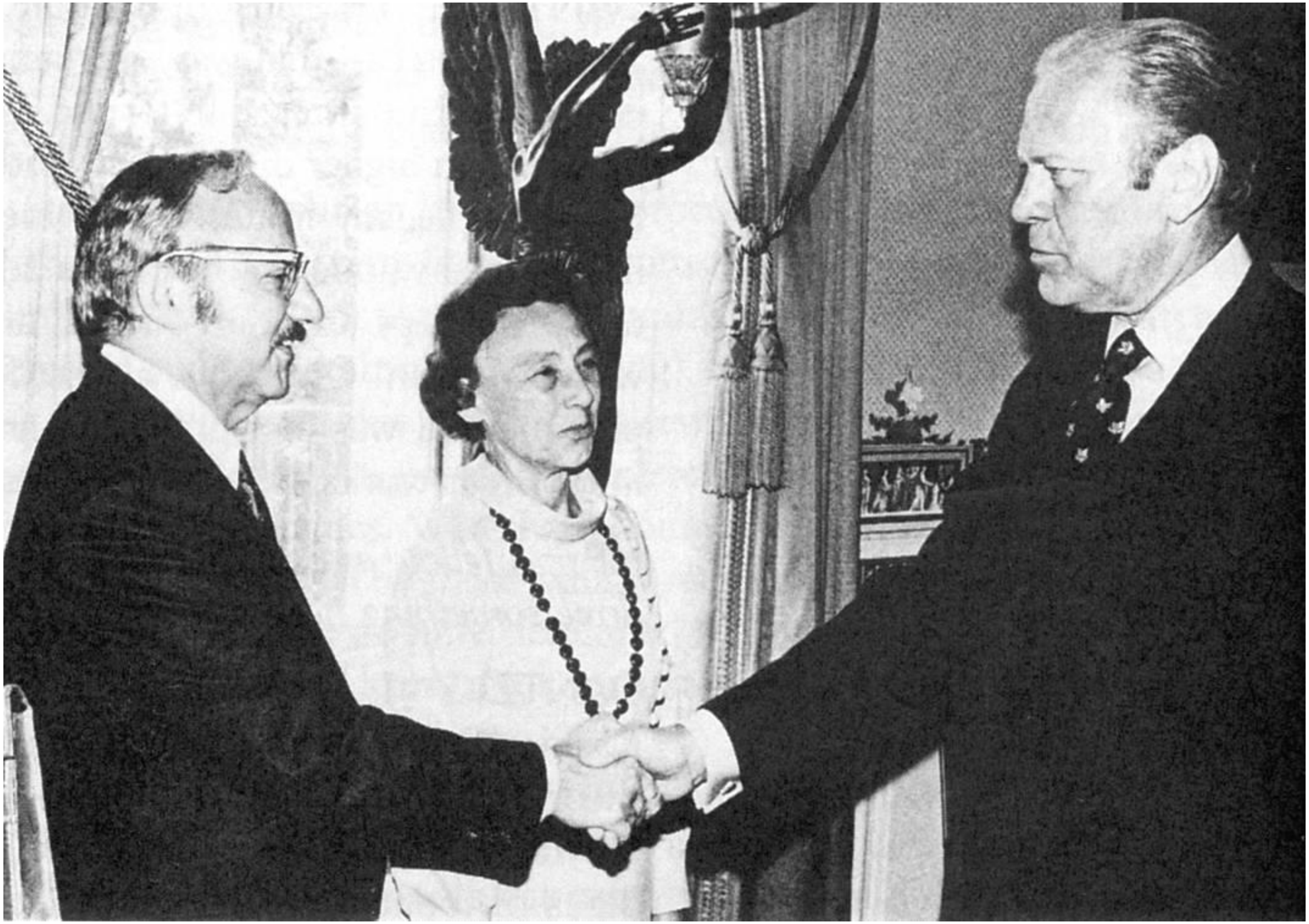
Linear Programming/Optimization: all functions are linear/affine

Linear Programming

- Why do we study LP's
  - Not just because solving non-linear problems are difficult
  - But because real-world problems are often formulated as linear equations and inequalities
    - Either because they indeed are linear
    - Or because it is unclear how to represent them and linear is an intuitive compromise
  - A stepping stone for solving more complicated nonlinear optimization problems, which you would see later.

LP, Nobel Prize,…

… and National Medal of Science



3 Main Categories in Optimization Covered in this Course

- **Linear Optimization** (Programming)
  - Search Algorithms
  - Simplex and Interior-Point Algorithms
- **Unconstrained Nonlinear Optimization**
  - 1st order methods, gradient method
  - 2nd order methods, Newton
- **Constrained Nonlinear Optimization**
  - 1st order, gradient projection, sequential LP, etc.
  - 2nd order, sequential Newton
  - Lagrangian Relaxation, Primal-Dual, etc.

Other Classifications:

Quadratic, Convex, Integer, Mixed-Integer, Binary, etc.

Issues in Optimization

- **Problem Size**
  - Small – by hand
  - Medium – by software
  - Large –by decomposition
- **Algorithm Complexity**
  - Convergence speed
  - Local Convergence speed
- **Insight more than just the solution**?
  - Solution structure properties
  - Sensitivity analysis
  - Alternate formulations

What do you learn?

- Models – the Art: intuition and common sense
  - How formulate real problems using quantitative models
- Theory – the Science: theorems, geometries and universal rules
  - Necessary and Sufficient Conditions that must be true for the optimality of different classes of problems.
- Algorithms – the Engineering: algorithms, methodologies and software tools
  - How we apply the theory to robustly and efficiently solve problems and gain insight beyond the solution.
- Applications – AI, Machine Learning and Data Science
  - Logistic Regression, SVM, the Wasserstain barycenter, Reinforced learning/MDP, Information market,…

Art of Modeling, Formulation & Vocabulary

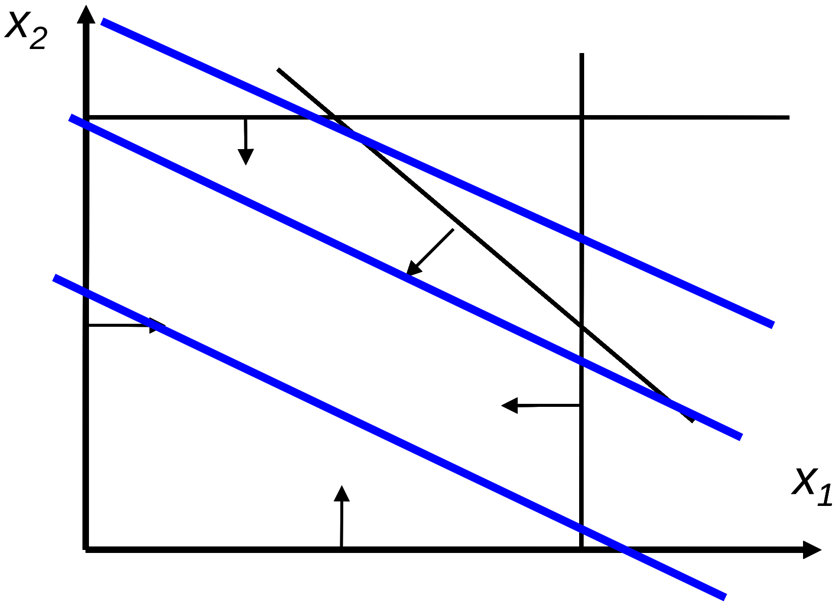
- Decision Variables $\boldsymbol{x} \in R^n$, yet to be decide
- Data/Coefficients, $\boldsymbol{c} \in R^n$, that are given and fixed
- Objective inner product $f = \boldsymbol{c}^T\boldsymbol{x}$: $R^n \rightarrow R$
- Constraint Set $X \subset R^n$
- Feasible solution $\boldsymbol{x} \in X$
- Optimal solution $\boldsymbol{x}^* \in X^*$
- Optimal value $z^* = f(\boldsymbol{x}^*)$

LP Example 1: Resource Allocation/Production Management

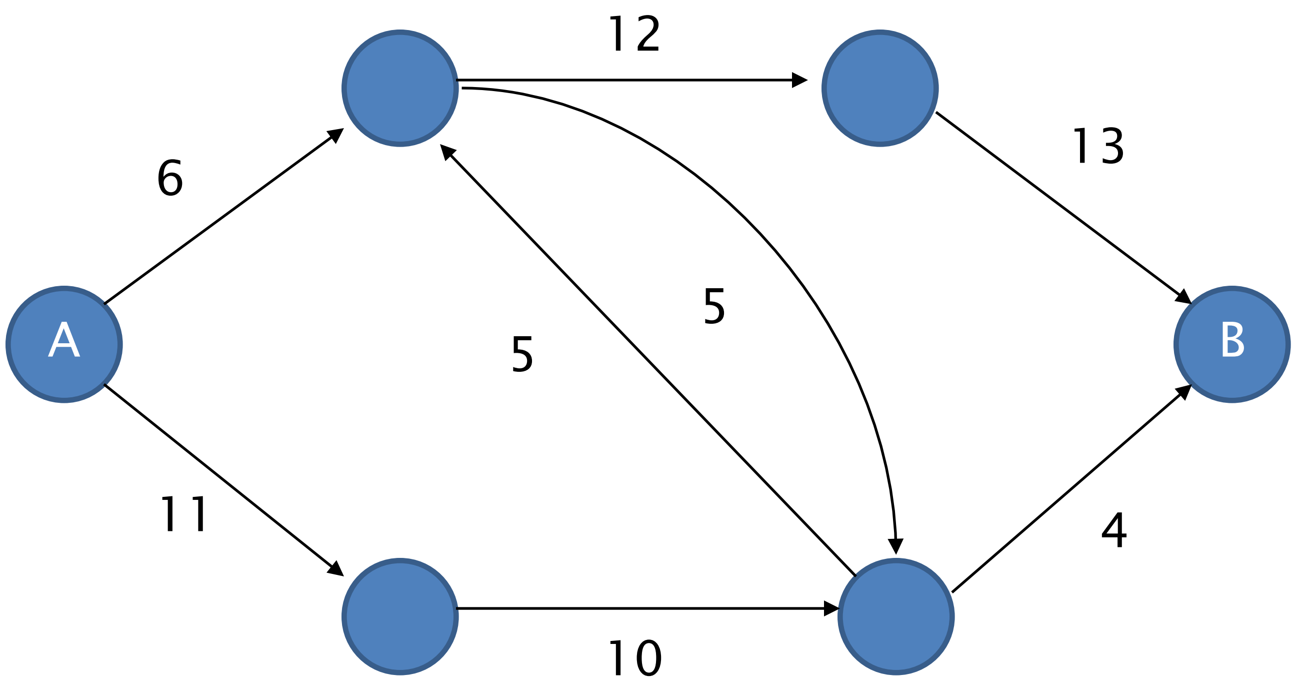
The Wyndor Glass Co. is a producer of high-quality glass products. It has three plants. Aluminum frames and hardware are made in Plant 1, wood frames are made in Plant 2, and Plant 3 is used to produce glass and assemble the products. Wyndor produces two products which require the resources of the three plants as follows:

Plant	Aluminum	Wood	Resources
1	1	0	100
2	0	2	200
3	1	1	150
Unit Profit	\$1000	\$2000	

$$
\begin{array}{ll}
\max & x_1 + 2x_2 \\
\text{s.t.} & x_1 \le 1, \\
& 2x_2 \le 2, \\
& x_1 + x_2 \le 1.5, \\
& x_1, x_2 \ge 0
\end{array}
$$

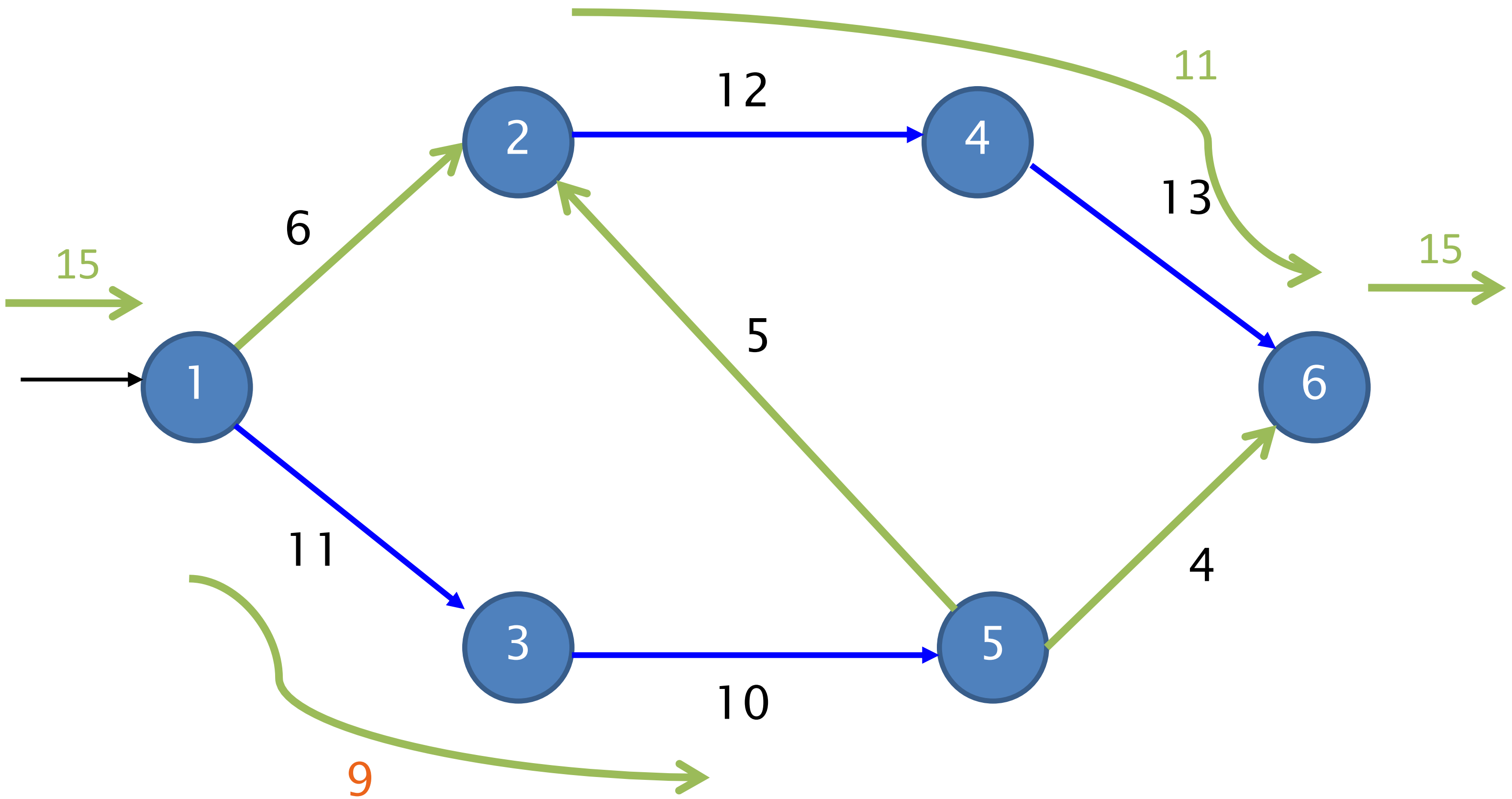


LP Example 2: Maximum Flow



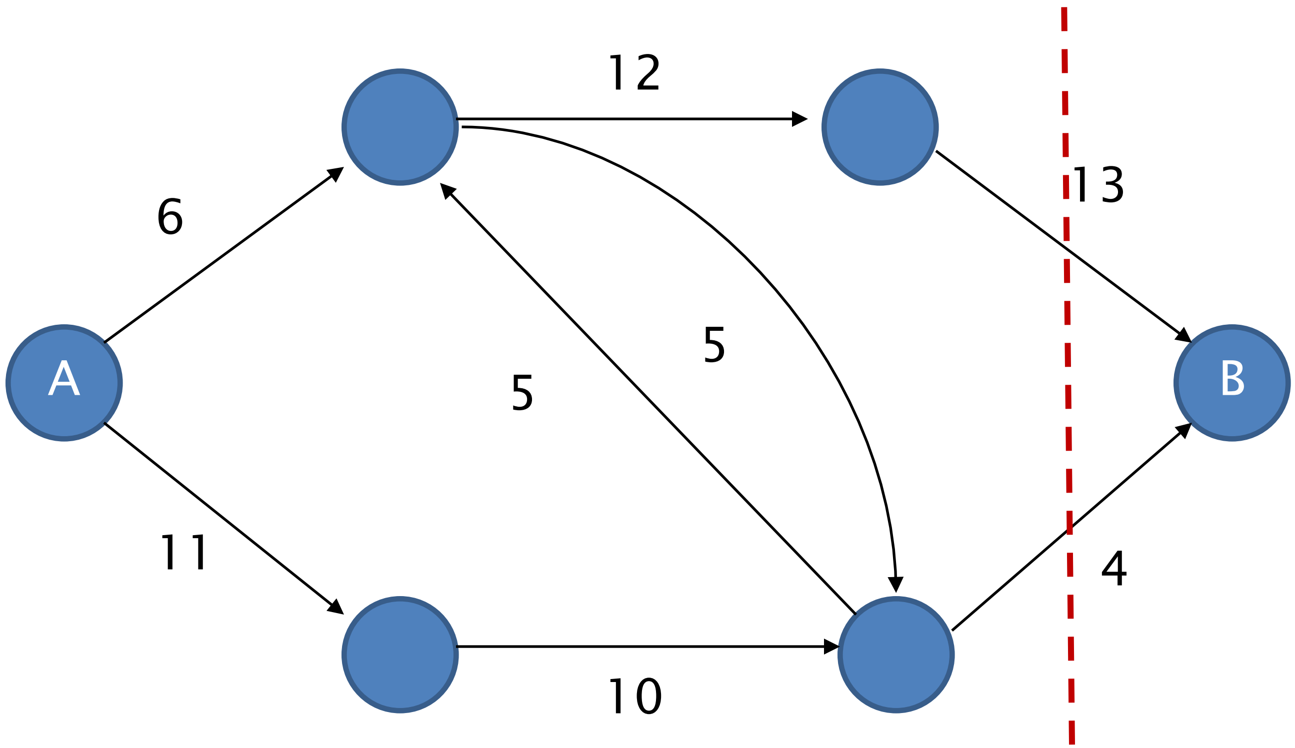
How much flow can travel from A to B, given that each of the directed connecting routes have flow limits/capacities?

Maximum Flow by Inspection



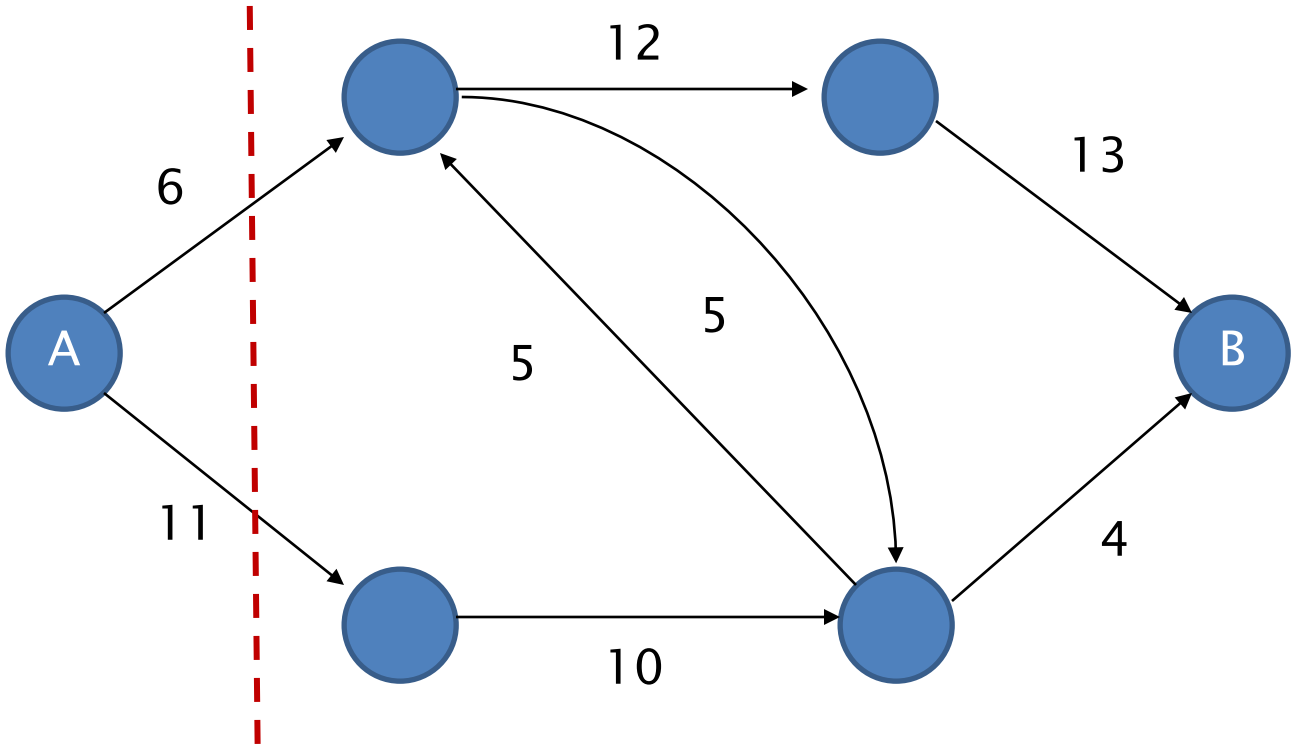
How to certify that 15 is maximal?

Cut in Maximum Flow I



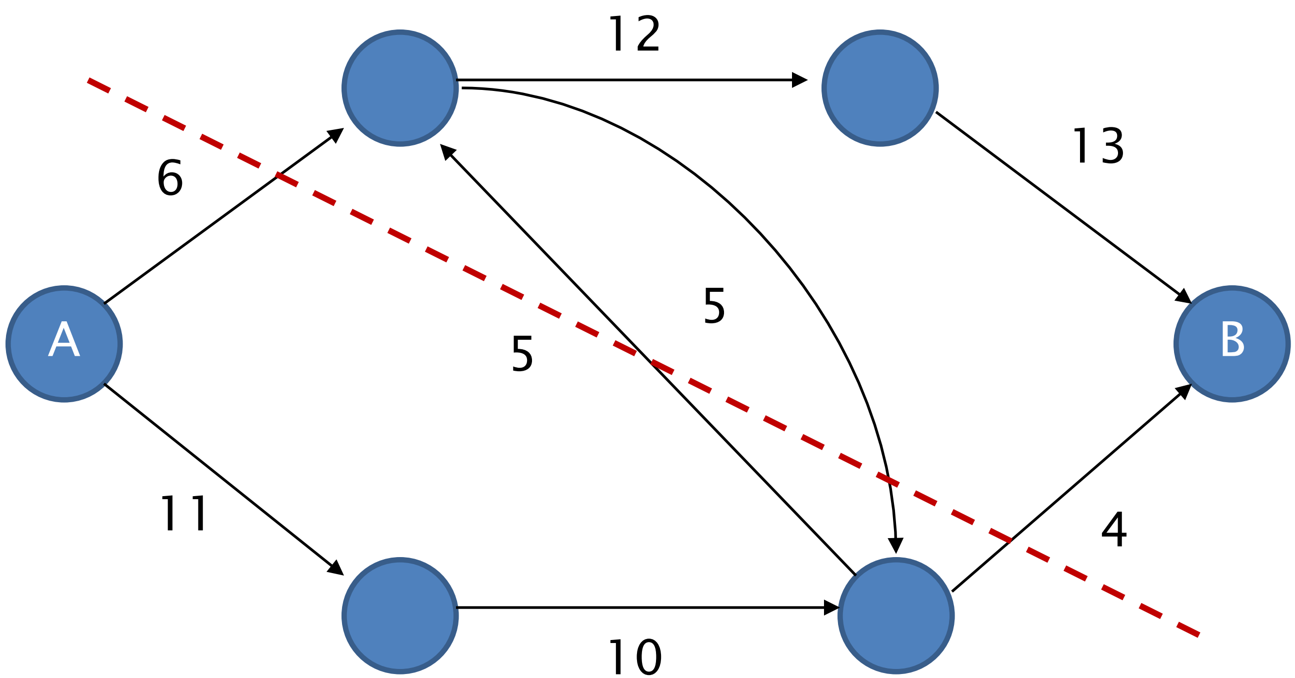
Cut value from Source site to Sink site=17

Cut in Maximum Flow II



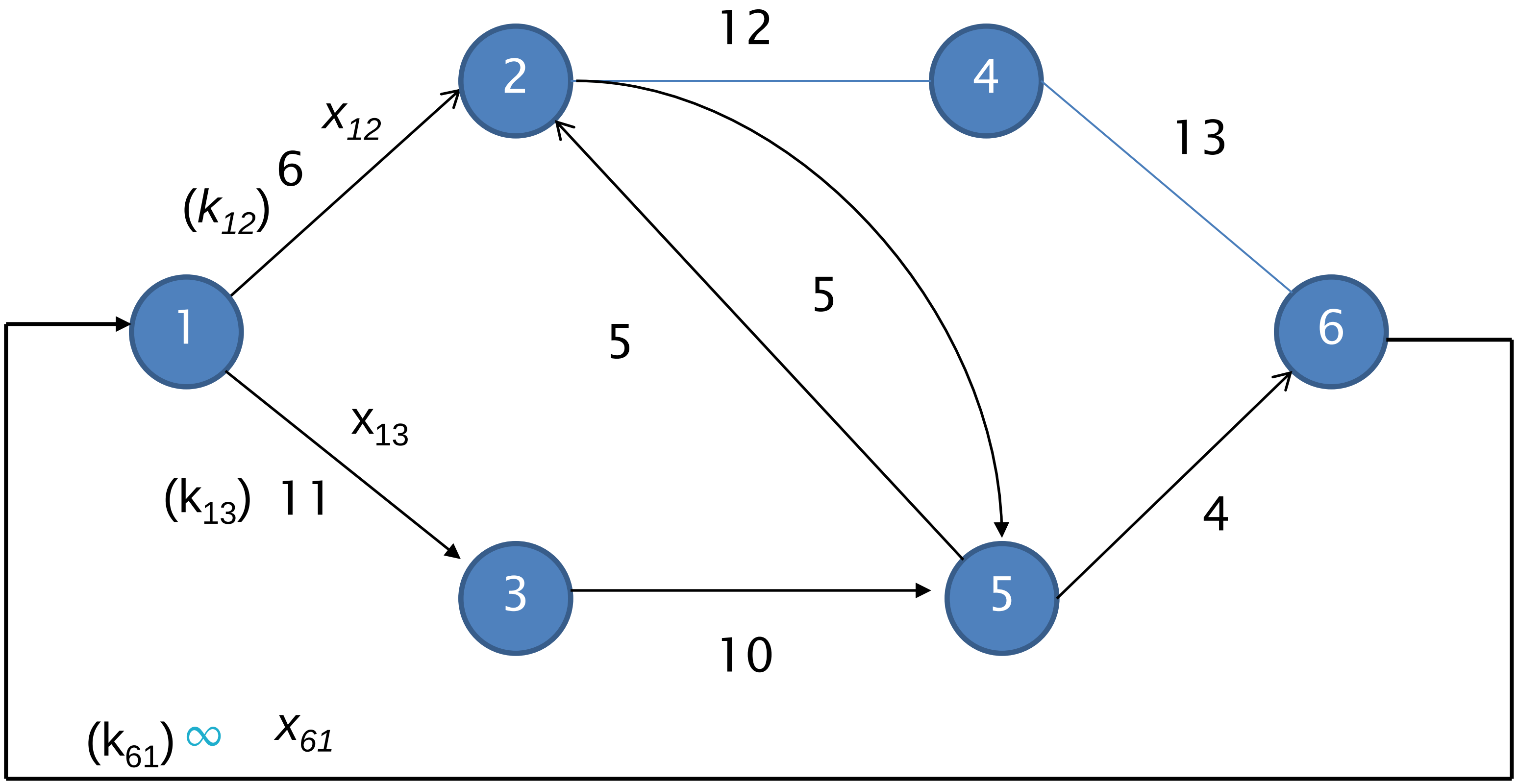
Cut value from Source site to Sink site=17

Cut in Maximum Flow III



Cut value from Source site to Sink site=15

Data points classification application in Machine Learning and Data Science



$$\max \quad x_{61}$$

$$\text{s.t.} \quad \sum_k x_{ki} = \sum_j x_{ij} \quad \forall\, i = 1,2,3,4,5,6$$

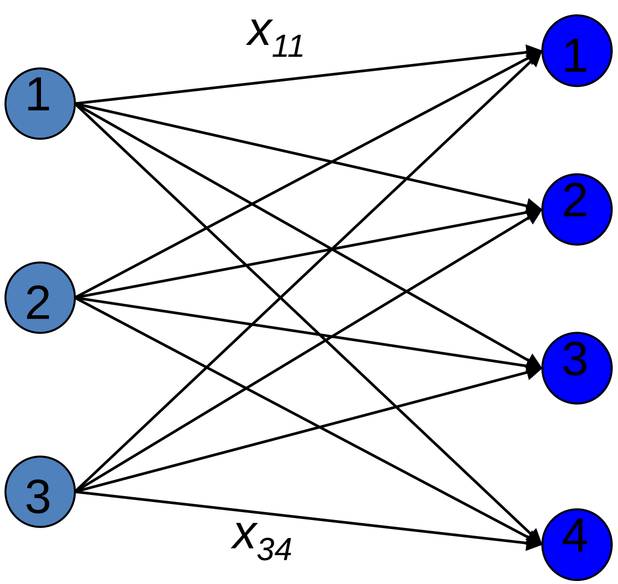
$$0 \le x_{ij} \le k_{ij} \quad \forall\, i, j = 1,2,3,4,5,6$$

inflow

outflow

LP Example 3: Transportation and Assignment

	Retailer 1	Retailer 2	Retailer 3	Retailer 4	SUPPLY
Warehouse 1	12 (c_{11})	13	4	6	500 (s_1)
Warehouse 2	6	4	10	11	700 (s_2)
Warehouse 3	10	9	12	14 (c_{34})	800 (s_3)
DEMAND	400 (d_1)	900 (d_2)	200 (d_3)	500 (d_4)	2000 (s_4)



$$
\begin{aligned}
\min \quad & \sum_{i=1}^{3}\sum_{j=1}^{4} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{4} x_{ij} = s_i, \quad \forall\, i = 1,2,3 \\
& \sum_{i=1}^{3} x_{ij} = d_j, \quad \forall\, j = 1,2,3,4 \\
& x_{ij} \geq 0, \quad \forall\, i, j
\end{aligned}
$$

Abstract Model

Inventory Planning: ***s*** is part of the decision vars.

Machine Learning: The Wassestein Barycenter Problem I

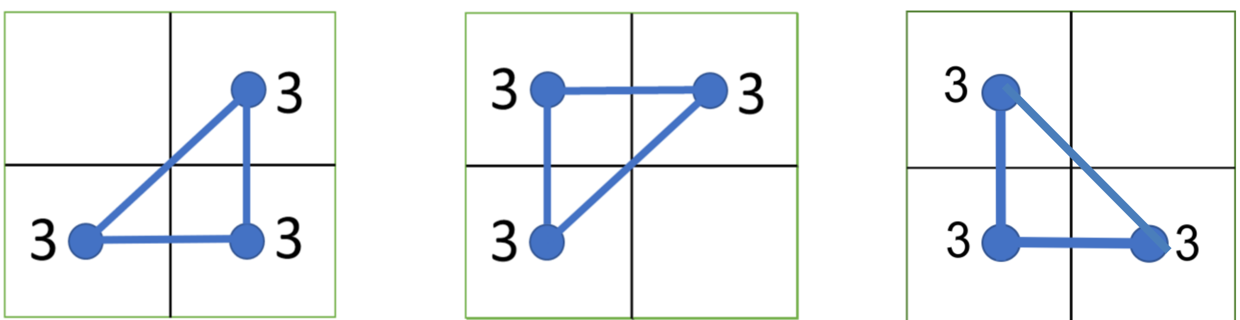
The minimal transportation cost in Data Science is called the Wasserstein distance between a supply distribution and a demand distribution.

The **Wasserstein Barycenter Problem** is to find a distribution/points such that the sum of its Wasserstein distances to each of a set of distributions/points would be minimized

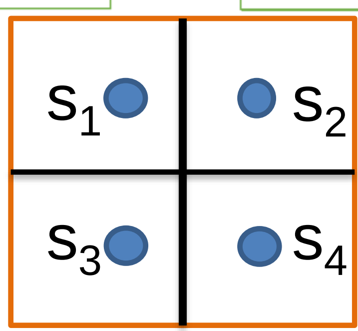
$$\min_s \sum_k \textbf{WD}(s, d^k) \text{ s.t. total mass constraint}$$

WD(s, d^k)=

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{N}\sum_{j=1}^{N} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{N} x_{ij} = s_i, \quad \forall\ i = 1, \dots, N \\
& \sum_{i=1}^{N} x_{ij} = d_j, \quad \forall\ j = 1, \dots, N \\
& x_{ij} \ge 0, \qquad \forall\ i, j
\end{aligned}
$$



Three possible demand distribution scenario of 4 cities



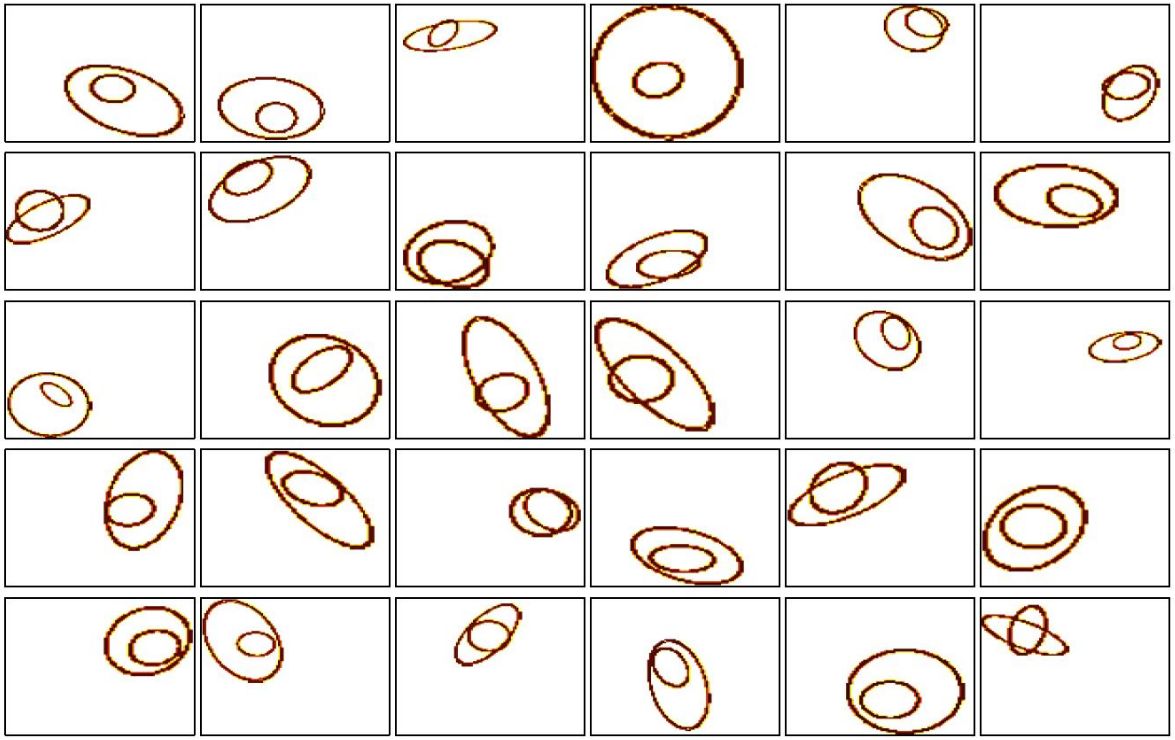
Constraints:

$s_1+s_2+s_3+s_4=9$

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

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

Machine Learning: The Wassestein Barycenter Problem II



What is the best ``mean or consensus‘’ image from a set of images (pixel distributions)?

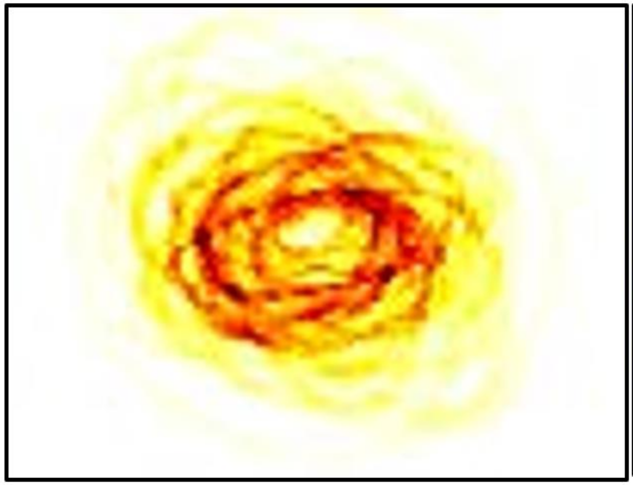
- Simple average
- Simple average after re-centering
- The Wasserstein Barycenter of the set of images (self re-center and rotation)

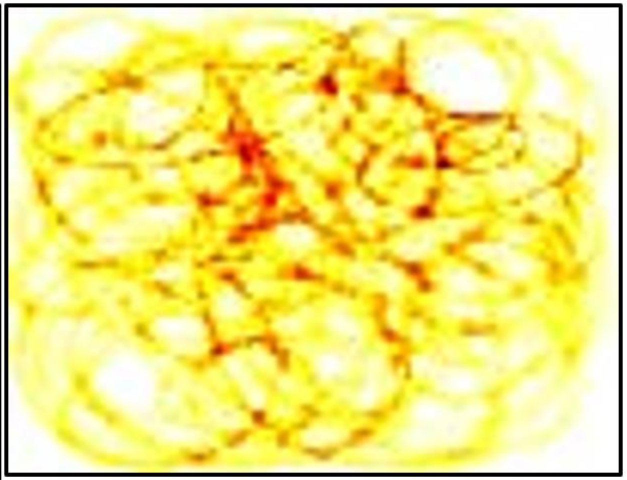
Machine Learning: The Wassestein Barycenter Problem III

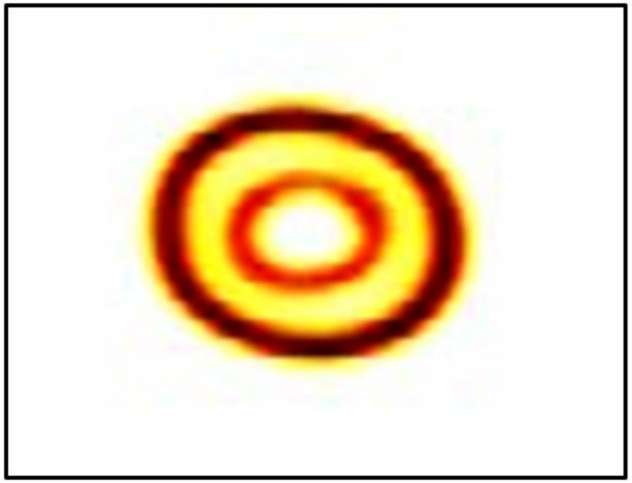
The simple avarage of n points is

$$\mathbf{s}=(\sum_k \mathbf{d}^k)/n \quad \text{or} \quad \min_{\mathbf{s}} \sum_{\mathbf{k}} (\|\mathbf{s}-\mathbf{d}^k\|_2)^2$$

The **Wasserstein Barycenter Problem** is to find a distribution/points such that the sum of its Wasserstein distances to each of a set of distributions/points would be minimized (self re-center and rotation).





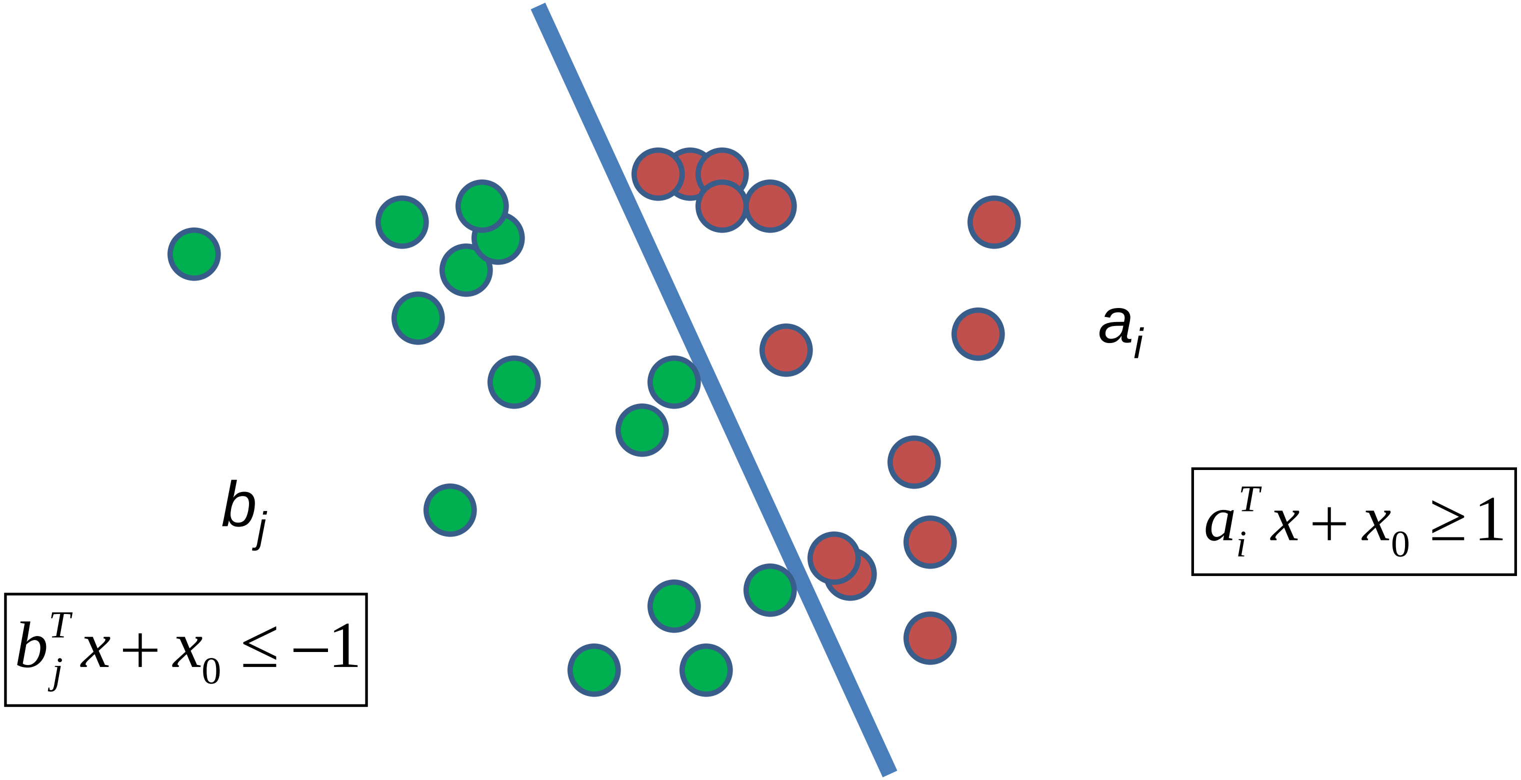


Simple average after re-centering

Simple average

the Barycenter image

LP Example 4: Support Vector Machine



$$\{y:\ y^T x + x_0 = 0\}$$

x is the normal direction or slope vector and x_0 is the intersect

Find a line to **strictly** separate greens and reds

LP Example 4: Is Strict Separation Possible?

$$a_i^T x + x_0 > 0, \forall i$$
$$b_j^T x + x_0 < 0, \forall j$$

Are there x and x_0 such that the following (open) inequalities are all satisfied

$$a_i^T x + x_0 \geq \varepsilon, \forall i$$
$$b_j^T x + x_0 \leq -\varepsilon, \forall j$$

Are there x and x_0 such that the following inequalities are all satisfied for arbitrarily small ε.

$\Updownarrow$

$$a_i^T x + x_0 \geq 1, \forall i$$
$$b_j^T x + x_0 \leq -1, \forall j$$

Divide x and x_0 by ε., the problem can be equivalently reformulated.

This is a special LP, called linear feasibility problem.

LP Example 4: Electric Vehicle Charging Schedule and Inventory Control

	Period 1	Period 2	Period 3	Period 4	Period 5
Price ($)	1.25 (c_1)	1.35 (c_2)	1.25 (c_3)	1.10 (c_4)	1.05 (c_5)
Demand (kw)	60 (d_1)	110 (d_2)	100 (d_3)	40 (d_4)	0 (d_5)
Charging (kw)	x_1	x_2	x_3	x_4	x_5
Inventory (I_0)	I_1	I_2	I_3	I_4	I_5

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{5} c_i x_i \\
\text{s.t.} \quad & I_{i-1} + x_i - d_i = I_i, \quad \forall\, i = 1,2,3,4,5 \\
& I_{i-1} + x_i \le K, \quad \forall\, i = 1,2,3,4,5 \\
& x_i \ge 0, I_i \ge 0, \quad \forall\, i.
\end{aligned}
$$

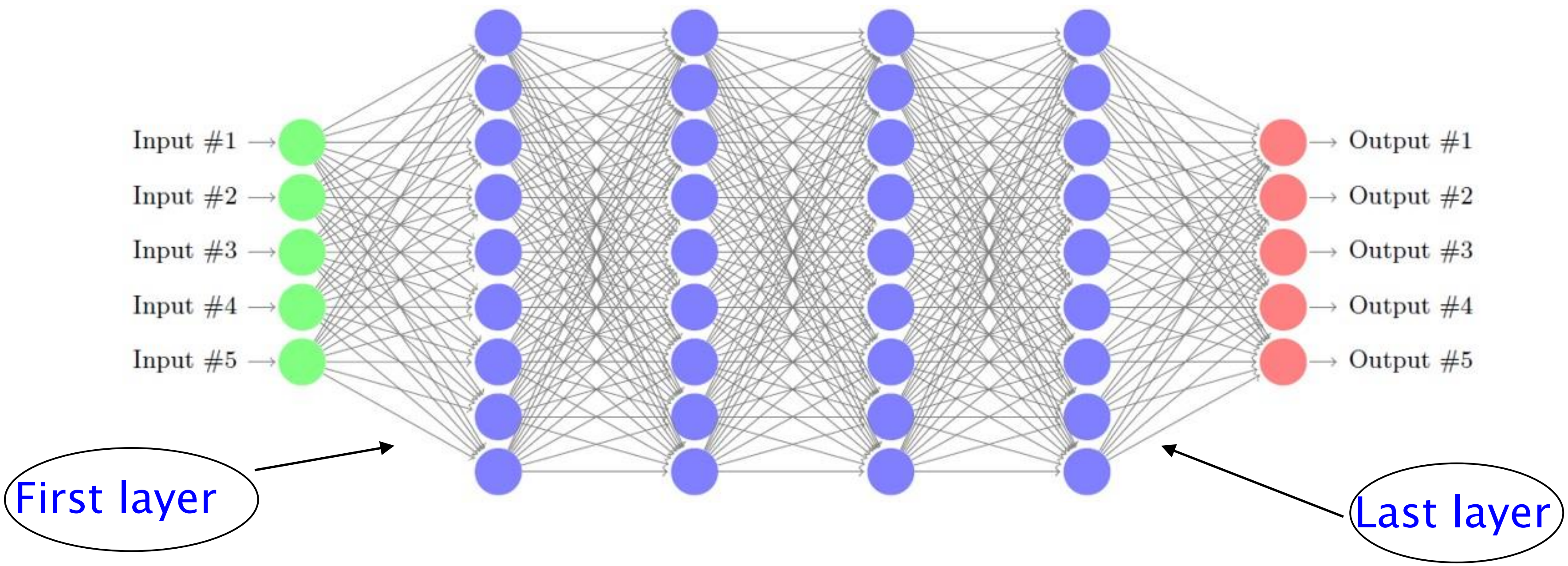
LP Example 5: When Discharge is Allowed

	Period 1	Period 2	Period 3	Period 4	Period 5
Price ($)	1.25 (c_1)	1.35 (c_2)	1.25 (c_3)	1.10 (c_4)	1.05 (c_5)
Demand (kw)	60 (d_1)	110 (d_2)	100 (d_3)	40 (d_4)	0 (d_5)
Charging (kw)	x_1	x_2	x_3	x_4	x_5
Inventory (I_0)	I_1	I_2	I_3	I_4	I_5

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{5} c_i x_i \\
\text{s.t.} \quad & I_{i-1} + x_i - d_i = I_i, \quad \forall\, i = 1,2,3,4,5 \\
& I_{i-1} + x_i \le K, \quad \forall\, i = 1,2,3,4,5 \\
& I_i \ge 0, \quad \forall\, i.
\end{aligned}
$$

Nonlinear Optimization: Bird or Plane?

Neural Network Design for Prediction



$$\text{optimize} \quad F(w_{i,j})$$

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

from a training set of pairs of inputs - outputs data so that when a new input data come the system predict what output would be.

Back to Linear Programming

$$\begin{array}{ll} \max & x_1 + 2x_2 \\ \text{s.t.} & x_1 \le 1, \\ & 2x_2 \le 2, \\ & x_1 + x_2 \le 1.5, \\ & x_1 \ge 0, x_2 \ge 0 \end{array}$$

$$\begin{array}{ll} \max & 1\cdot x_1 + 2\cdot x_2 \\ \text{s.t.} & 1\cdot x_1 + 0\cdot x_2 \le 1, \\ & 0\cdot x_1 + 2\cdot x_2 \le 2, \\ & 1\cdot x_1 + 1\cdot x_2 \le 1.5, \\ & x_1 \ge 0, x_2 \ge 0. \end{array}$$

$$\begin{array}{ll} \max & c_1 x_1 + c_2 x_2 \\ \text{s.t.} & a_{11} x_1 + a_{12} x_2 \le b_1 \\ & a_{21} x_1 + a_{22} x_2 \le b_2 \\ & a_{31} x_1 + a_{32} x_2 \le b_3 \\ & x_1 \ge 0, \; x_2 \ge 0.. \end{array}$$

Abstract Linear Programming Model

$$\max\ (\min) \quad c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

$$\text{s.t.} \quad a_{11} x_1 + a_{12} x_2 + \cdots + a_{1n} x_n \ \{\leq, =, \geq\}\ b_1$$

$$a_{21} x_1 + a_{22} x_2 + \cdots + a_{2n} x_n \ \{\leq, =, \geq\}\ b_2$$

$$\cdots \qquad \cdots$$

$$a_{m1} x_1 + a_{m2} x_2 + \cdots + a_{mn} x_n \{\leq, =, \geq\}\ b_m$$

$$x_1 \geq 0,\ x_2 \text{ free},\ \ldots,\ x_n \leq 0.$$

Input: $c_1, \ldots, c_2$, objective coef.; $b_1, \ldots, b_m$, constraint right-hand-side coef.

$a_{ij}, i = 1, \ldots, m; j = 1, \ldots, n$, constraint left-hand-side table or matrix coef.

Output: $x_1, \ldots, x_n$, decision v ariables

LP in Compact Matrix Form

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1\mathrm{n}} \\ a_{21} & a_{22} & \cdots & a_{2\mathrm{n}} \\ \cdots & \cdots & \cdots & \cdots \\ a_{\mathrm{m}1} & a_{\mathrm{m}2} & \cdots & a_{\mathrm{mn}} \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_m \end{pmatrix}, c = \begin{pmatrix} c_1 \\ c_2 \\ \cdots \\ c_n \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix}$$

Coefficient matrix

RHS vector

Obj. vector

decision vector

$$\begin{aligned} \max(\min) \quad & c^T x \\ \text{s.t.} \quad & A\,x\ \{\leq, =, \geq\}\ b, \\ & x\ \{\geq, \leq\}\ 0 \text{ or free.} \end{aligned}$$

Some Facts of Linear Programming

- Add a constant to the objective function does not change the optimality
- Scale the objective coefficients does not change the optimality
- Scale the right-hand-side coefficients does not change the optimality but the solution scaled accordingly
- Reorder the decision variables (together with their corresponding objective and constraint coefficients) does not change the optimality
- Reorder the constraints (together with their right-hand-side coefficients) does not change the optimality
- Multiply both sides of an equality constraint by a constant does not change the optimality
- Pre-multiply both sides of all equality constraints by a non-singular matrix does not change the optimality