Mathematical Optimization Models and Applications I

Yinyu Ye

Department of Management Science and Engineering

Stanford University

Stanford, CA 94305, U.S.A.

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

Chapters 1, 2.1-2, 6.1-2, 7.2, 11.3, 11.6

What you lean in MS&E314?

- Review mathematical optimization theories and algorithms for the Engineering disciplines.
- Introduce additional conic and nonlinear/nonconvex optimization/game algorithms comparing to MS&E310.
- Describe new/recent effective optimization/game models/methods/algorithms in Market Design, Data Science and Machine Learning.
- Emphasis is on nonlinear, nonconvex and stochastic/sample-based optimization methodologies and practices together with convex optimization algorithms.

Mathematical Optimization/Programming (MP)

The field of optimization is concerned with the study of maximization and minimization of mathematical functions. Very often the arguments of (i.e., variables or unknowns in) these functions are subject to side conditions or constraints. By virtue of its great utility in such diverse areas as applied science, engineering, economics, finance, medicine, and statistics, optimization holds an important place in the practical world and the scientific world. Indeed, as far back as the Eighteenth Century, the famous Swiss mathematician and physicist Leonhard Euler (1707-1783) proclaimed[a] that . . . nothing at all takes place in the Universe in which some rule of maximum or minimum does not appear.

[a] See Leonhardo Eulero, *Methodus Inviendi Lineas Curvas Maximi Minimive Proprietate Gaudentes*, Lausanne & Geneva, 1744, p. 245.

Mathematical Optimization/Programming Model

The class of mathematical optimization/programming problems considered in this course can all be expressed in the form

$$\begin{aligned} \text{(P)} \quad & \text{minimize} \quad f(\mathbf{x}) \\ & \text{subject to} \quad \mathbf{x} \in \mathcal{X} \end{aligned}$$

where $\mathcal{X}$ usually specified by constraints:

$$\begin{aligned} c_i(\mathbf{x}) &= 0 \quad i \in \mathcal{E} \\ c_i(\mathbf{x}) &\leq 0 \quad i \in \mathcal{I}. \end{aligned}$$

If the constraint functions are linear/affine type, then $\mathcal{X}$ is a convex polyhedral set/region.

Model Classifications

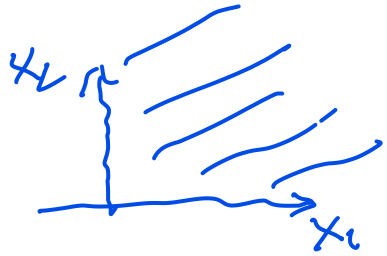
Optimization problems are generally divided into Unconstrained, Linear and Nonlinear Programming based upon the objective and constraints of the problem

- Unconstrained Optimization: Ω is the entire space R^n
- Linear Optimization: If both the objective and the constraint functions are linear/affine
- Nonlinear Optimization: If the objective/constraints contain general nonlinear functions
- Convex Optimization: If the objective is a convex function and the constraint region is a convex set
- Conic Linear Optimization: If both the objective and the constraint functions are linear/affine, and variables are in a convex cone.
- (Mixed) Integer Optimization: If the some variables are restricted to be integral
- Stochastic Optimization: Optimize the expected objective function with random parameters
- Fixed-Point or Min-Max Optimization: Optimization of multiple agents with zero-sum objectives

We present a few optimization examples in this lecture that we would cover through out this course.

Structured Optimization: Conic Linear Programming (CLP)

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \in K. \end{array}$$

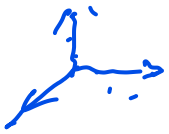


Linear Programming (LP): when K is the nonnegative orthant cone

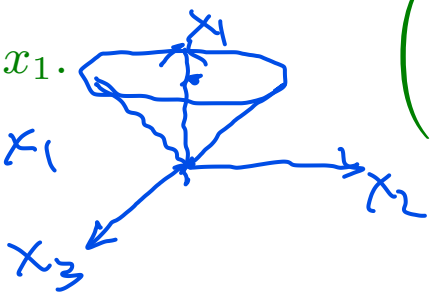
Second-Order Cone Programming (SOCP): when K is the second-order cone

Semidefinite Cone Programming (SDP): when K is the semidefinite matrix cone

$$\begin{array}{ll} \min & 2x_1 + x_2 + x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 = 1, \\ & (x_1; x_2; x_3) \geq \mathbf{0}; \end{array}$$

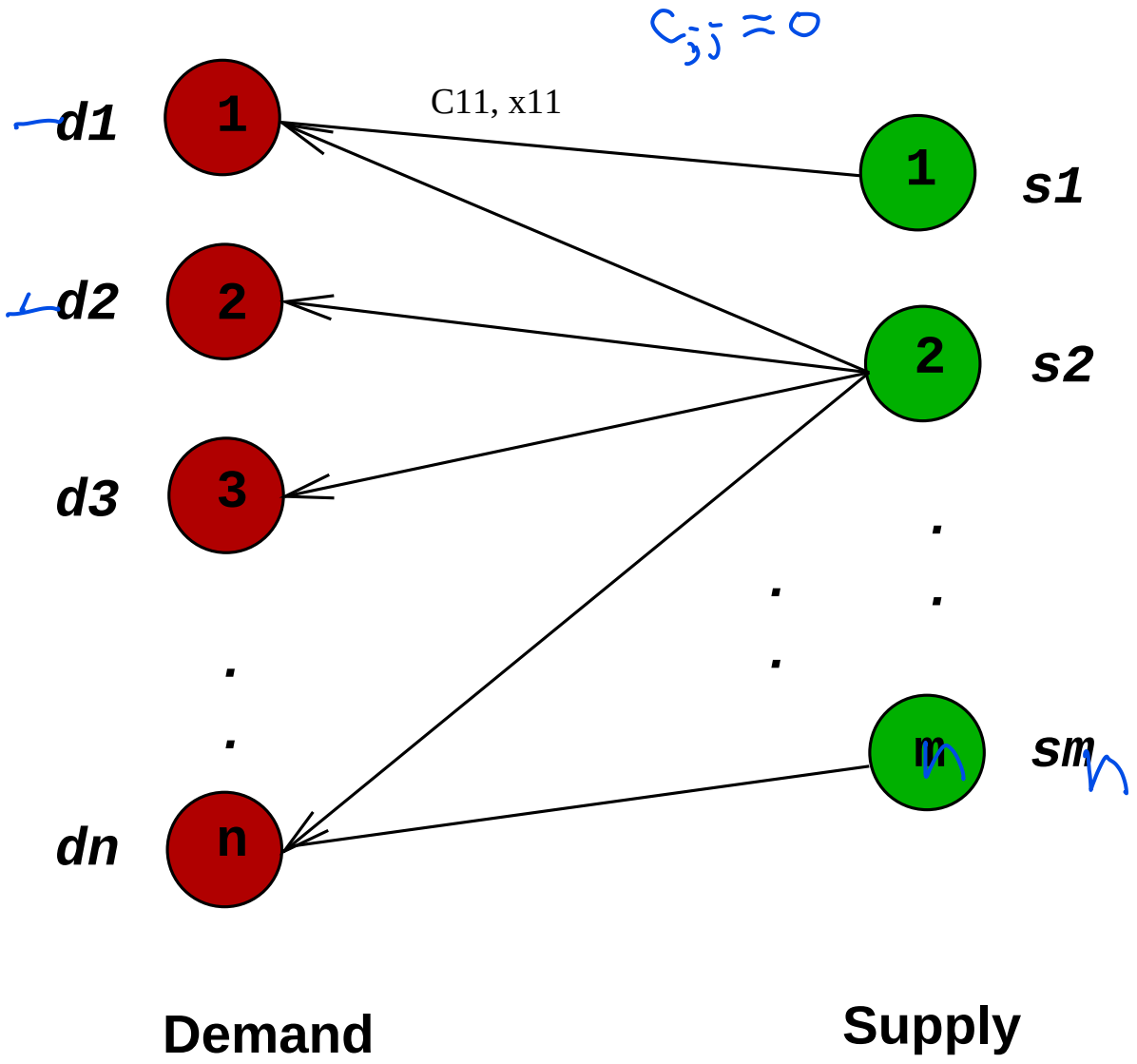


$$\begin{array}{ll} \min & 2x_1 + x_2 + x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 = 1, \\ & \sqrt{x_2^2 + x_3^2} \leq x_1. \end{array}$$



$$\begin{array}{ll} \min & 2x_1 + x_2 + x_3 \\ \text{s.t.} & x_1 + x_2 + x_3 = 1, \\ & \begin{pmatrix} x_1 & x_2 \\ x_2 & x_3 \end{pmatrix} \succeq \mathbf{0}, \end{array}$$

The Transportation Problem



Mathematical Optimization Model:

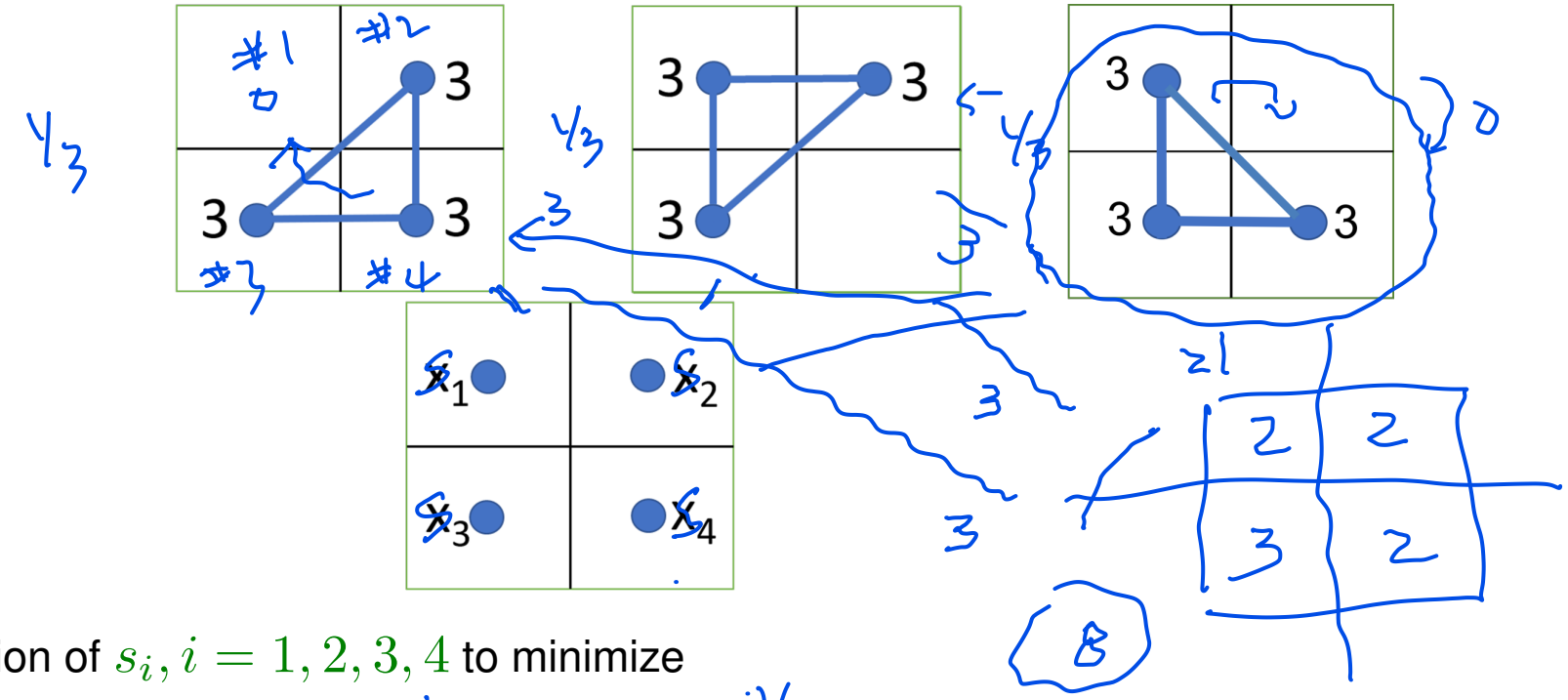
$$
\begin{aligned}
\min \quad & \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} && = s_i,\ \forall i = 1, ..., m \\
& \sum_{i=1}^{m} x_{ij} && = d_j,\ \forall j = 1, ..., n \\
& x_{ij} && \geq 0,\ \forall i, j.
\end{aligned}
$$

The minimal transportation cost is called the Wasserstein Distance (WD) between supply distribution $\mathbf{s}$ and demand distribution $\mathbf{d}$ (can be interpreted as two probability distributions after normalization). This is a linear program!

What happen if supplies $\mathbf{s}$ are also decision variables?

The Wasserstein Barycenter Problem is to find a distribution such that the sum of its Wasserstein Distance to each of a set of distributions would be minimized.

A Wassestein Barycenter Application: Stochastic Optimization



Find distribution of $s_i, i = 1, 2, 3, 4$ to minimize

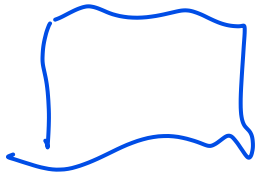
$$\min \frac{1}{3} WD_l(\mathbf{s}, \mathbf{d}_l) + WD_m(\mathbf{s}, \mathbf{d}_m) + \frac{2}{3} WD_r(\mathbf{s}, \mathbf{d}_r)$$

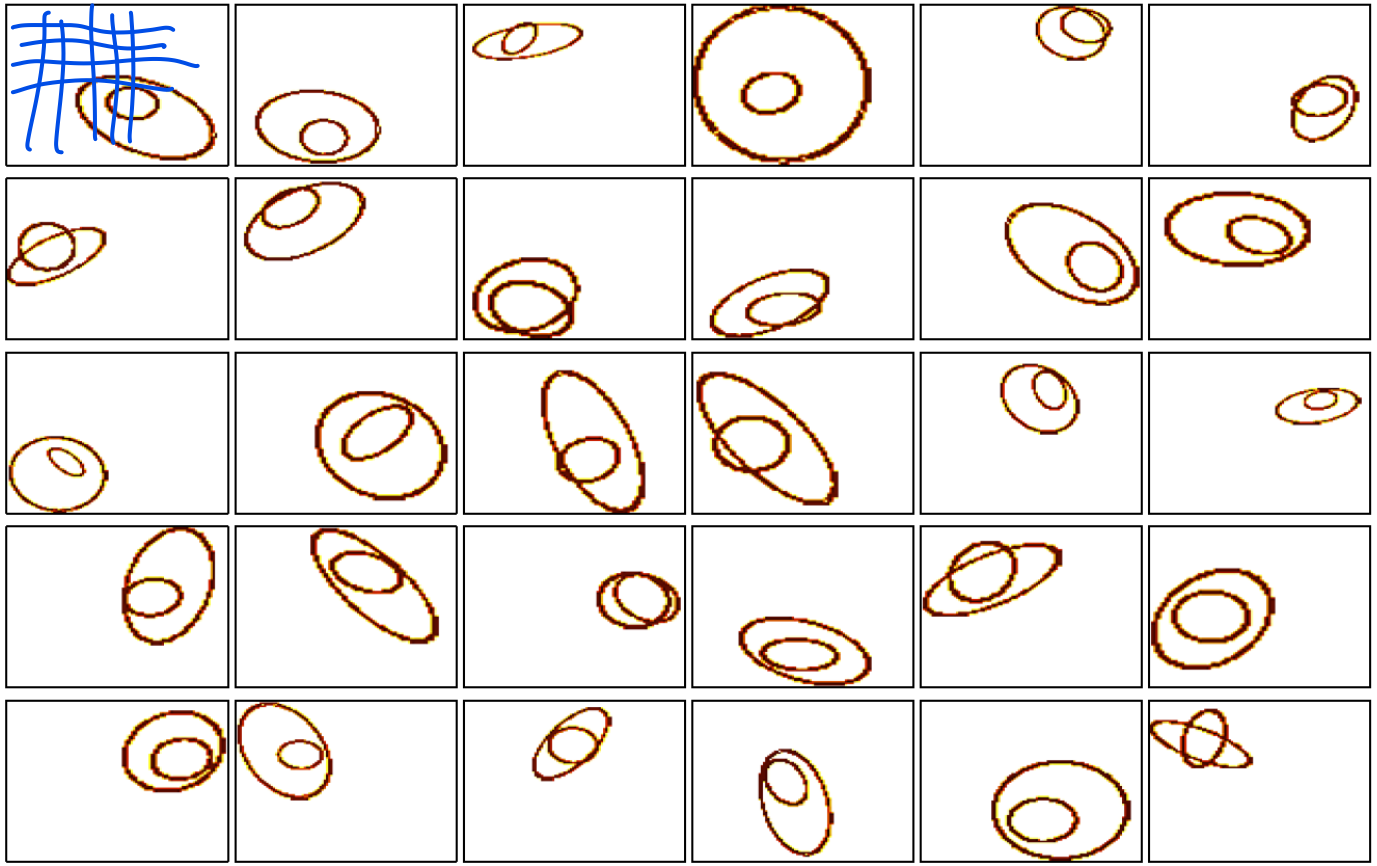
$$\text{s.t.} \qquad s_1 + s_2 + s_3 + s_4 = 9, \qquad s_i \geq 0,\ i = 1, 2, 3, 4.$$

The objective is a nonlinear function, but its gradient vector $\nabla WD_l(\mathbf{s}, \mathbf{d}_l)$, $\nabla WD_m(\mathbf{s}, \mathbf{d}_m)$ and $\nabla WD_r(\mathbf{s}, \mathbf{d}_r)$ are shadow prices of the three sub-transportation problems –popularly used in Hierarchy Optimization.

The Wasserstein Barycenter (Mean) Problem in Data Science

What is the "mean or consensus" image from a set of images/distributions:





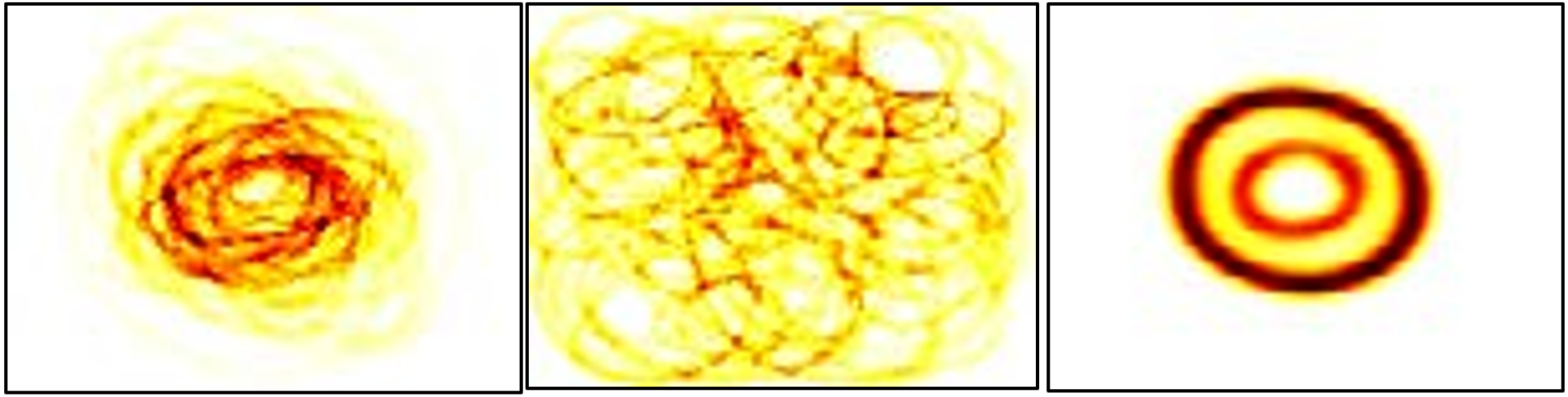


Figure 1: Mean picture constructed from the (a) Euclidean mean after re-centering images (b) Euclidean mean (c) Wasserstein Barycenter (self recenter, resize and rotate)

Euclidean Mean/Center:

$$\mathbf{x} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{a}_i, \quad \text{or} \quad \min_{\mathbf{x}} \sum_{i=1}^{n} \|\mathbf{x} - \mathbf{a}_i\|_2^2,$$

which is an unconstrained optimization, or least-squares, problem

Algorithmic Market Design: World Cup Prediction Market

Order:	#1	#2	#3	#4	#5
Argentina	1	0	1	1	0
Brazil	1	0	0	1	1
Italy	1	0	1	1	0
Germany	0	1	0	1	1
France	0	0	1	0	0
Bidding Prize:π	0.75	0.35	0.4	0.95	0.75
Quantity limit:$\mathbf{q}$	10	5	10	10	5
Order fill:$\mathbf{x}$	x_1	x_2	x_3	x_4	x_5

Prediction Call Auction Market

Given m potential states that are mutually exclusive and exactly one of them will be realized at the maturity.

An order is a bet on one or a combination of states, with a price limit (the maximum price the participant is willing to pay for one unit of the order) and a quantity limit (the maximum number of units or shares the participant is willing to accept).

A contract on an order is a paper agreement so that on maturity it is worth a notional $\$1$ dollar if the order includes the winning state and worth $\$0$ otherwise.

There are n orders submitted now.

Prediction Call Auction Market: Input Order Data

The ith order is given as $(\mathbf{a}_{i\cdot} \in R_+^m,\ \pi_i \in R_+,\ q_i \in R_+)$: $\mathbf{a}_{i\cdot}$ is the betting indication row vector where each component is either 1 or 0

$$\mathbf{a}_{i\cdot} = (a_{i1},\ a_{i2},\ ...,\ a_{im})$$

where 1 is winning state and 0 is non-winning state; π_i is the price limit for one unit of such a contract, and q_i is the maximum number of contract units the better like to buy.

Prediction Call Auction Market: Output Order-Fill Decisions

Let x_i be the number of units or shares awarded to the ith order. Then, the ith bidder will pay the amount $\pi_i \cdot x_i$ and the total amount collected would be $\pi^T \mathbf{x} = \sum_i \pi_i \cdot x_i$.

If the jth state is the winning state, then the auction organizer need to pay the winning bidders

$$\left(\sum_{i=1}^{n} a_{ij} x_i \right) = \mathbf{a}_{\cdot j}^T \mathbf{x}$$

where column vector

$$\mathbf{a}_{\cdot j} = (a_{1j};\ a_{2j};\ ...;\ a_{nj})$$

The question is, how to decide $\mathbf{x} \in R^n$, that is, how to fill the orders.

Prediction Call Auction Market: Worst-Case Profit Maximization

$$\begin{aligned} \max \quad & \pi^T \mathbf{x} - \max_j \{\mathbf{a}_{\cdot j}^T \mathbf{x}\} \\ \text{s.t.} \quad & \mathbf{x} \leq \mathbf{q}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

$$\begin{aligned} \max \quad & \pi^T \mathbf{x} - \max(A^T \mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \leq \mathbf{q}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

This is NOT a linear program.

Prediction Call Auction Market: Linear Optimization Representation

However, the problem can be rewritten as

$$\begin{array}{lll} \max & \pi^T \mathbf{x} - z & \\ \text{s.t.} & A^T \mathbf{x} - \mathbf{e} \cdot z & \leq \mathbf{0}, \\ & \mathbf{x} & \leq \mathbf{q}, \\ & \mathbf{x} & \geq \mathbf{0}, \end{array}$$

where $\mathbf{e}$ is the vector of all ones. This is a linear program.

Call Auction

$$\begin{array}{lll} \max_{x,s,z} & \pi^T \mathbf{x} - z & \\ \text{s.t.} & A^T \mathbf{x} - \mathbf{e} \cdot y + \mathbf{s} & = \mathbf{0}, \\ & \mathbf{x} & \leq \mathbf{q}, \\ & (\mathbf{x}, \mathbf{s}) & \geq \mathbf{0}, \quad z \text{ free}, \end{array}$$

Prediction Online Auction Market

In real applications, data/information is revealed sequentially, and one has to make decisions sequentially based on what is known – cannot wait for solving the offline problem.

$$\begin{aligned} \max \quad & \pi^T \mathbf{x} - z + u(\mathbf{s}) \\ \text{s.t.} \quad & A\mathbf{x} - \mathbf{e} \cdot z + \mathbf{s} = \mathbf{0}, \\ & \mathbf{x} \leq \mathbf{q}, \\ & \mathbf{x},\ \mathbf{s} \geq 0, \quad z \text{ free}, \end{aligned}$$

$u(\mathbf{s})$: a value function for the market organizer on slack shares.

$\mathbf{s}^0$: initial seed of slack shares.

If $u(\cdot)$ is a strictly concave function, then the state price vector is unique.

Online Market: solve the convex problem sequentially - SCPM.

A Physical Auction Market

	order 1($t=1$)	order 2($t=2$)	.....	Inventory($\mathbf{b}$)
Price(π_t)	\$100	\$30	...	
Decision	x_1	x_2	...	
Pants	1	0	...	100
Shoes	1	0	...	50
T-shirts	0	1	...	500
Jacket	0	0	...	200
Socks	1	1	...	1000

Physical Auction Market: Linear Optimization Representation

$$\begin{aligned} \text{maximize}_{\mathbf{x}} \quad & \textstyle\sum_{j=1}^{n} \pi_j x_j \\ \text{s.t.} \quad & \textstyle\sum_{j=1}^{n} a_{ij} x_j \le b_i, \ \forall\, i = 1, 2, ..., m, \\ & 0 \le x_j \le 1, \ \forall\, j = 1, ..., n. \end{aligned}$$

or online version

$$\begin{aligned} \text{(SCPM):} \quad \text{maximize}_{x_k, \mathbf{s}} \quad & \pi_k x_k + u(\mathbf{s}) \\ \text{s.t.} \quad & a_{ik} x_k + s_i = b_i - \textstyle\sum_{j=1}^{k-1} a_{ik} \bar{x}_k, \ \forall\, i = 1, 2, ..., m, \\ & 0 \le x_k \le 1, \\ & s_i \ge 0, \ \forall\, i = 1, ..., m. \end{aligned}$$

Physical Auction Market: Nonlinear or Non-Convex

$$\begin{aligned} \text{maximize}_{\mathbf{x}} \quad & \sum_{j=1}^{n} u_j(x_j) \\ \text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \ \forall\, i = 1, 2, ..., m, \\ & 0 \le x_j \le 1, \ \forall\, j = 1, ..., n. \end{aligned}$$

and/or

$$\begin{aligned} \text{maximize}_{\mathbf{x}} \quad & \sum_{j=1}^{n} u_j(x_j) \\ \text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \ \forall\, i = 1, 2, ..., m, \\ & x_j \in \{0, 1\}, \ \forall\, j = 1, ..., n. \end{aligned}$$

Non-Auction Market: Fisher and Arrow-Debreu Markets

The Fisher Buyer-Market:

$$\begin{aligned} \text{maximize}_{\mathbf{x}} \quad & \sum_{j=1}^{n} w_j \log(\mathbf{u}_j^T \mathbf{x}_j) \\ \text{s.t.} \quad & \sum_{j=1}^{n} \mathbf{x}_j = \mathbf{b} \\ & 0 \le \mathbf{x}_j, \ \forall\, j = 1, ..., n. \end{aligned}$$

The Arrow-Debreu Exchange-Market:

$$\begin{aligned} \text{maximize}_{\mathbf{x}} \quad & \sum_{j=1}^{n} u_j(\mathbf{x}_j) \\ \text{s.t.} \quad & \sum_{j=1}^{n} \mathbf{x}_j = \mathbf{0} \\ & \mathbf{v}_j \le \mathbf{x}_j \le \mathbf{u}_j, \ \forall\, j = 1, ..., n. \end{aligned}$$

Facility Location Problem

Let $\mathbf{c}_j$ be the location of client $j = 1, 2, ..., m$, and $\mathbf{y}$ be the location decision of a facility to be built. Then we solve

$$\text{minimize}_{\mathbf{y}} \quad \sum_j \|\mathbf{y} - \mathbf{c}_j\|_p.$$

Or equivalently (?)

$$\begin{aligned} \text{minimize} \quad & \sum_j \delta_j \\ \text{subject to} \quad & \mathbf{y} + \mathbf{x}_j = \mathbf{c}_j, \ \|\mathbf{x}_j\|_p \le \delta_j, \ \forall j. \end{aligned}$$

This is a p-order conic linear program (POCP) for $p \ge 1$.

In particular, when $p = 2$, it is an SOCP problem.

For simplicity, consider $m = 3$.

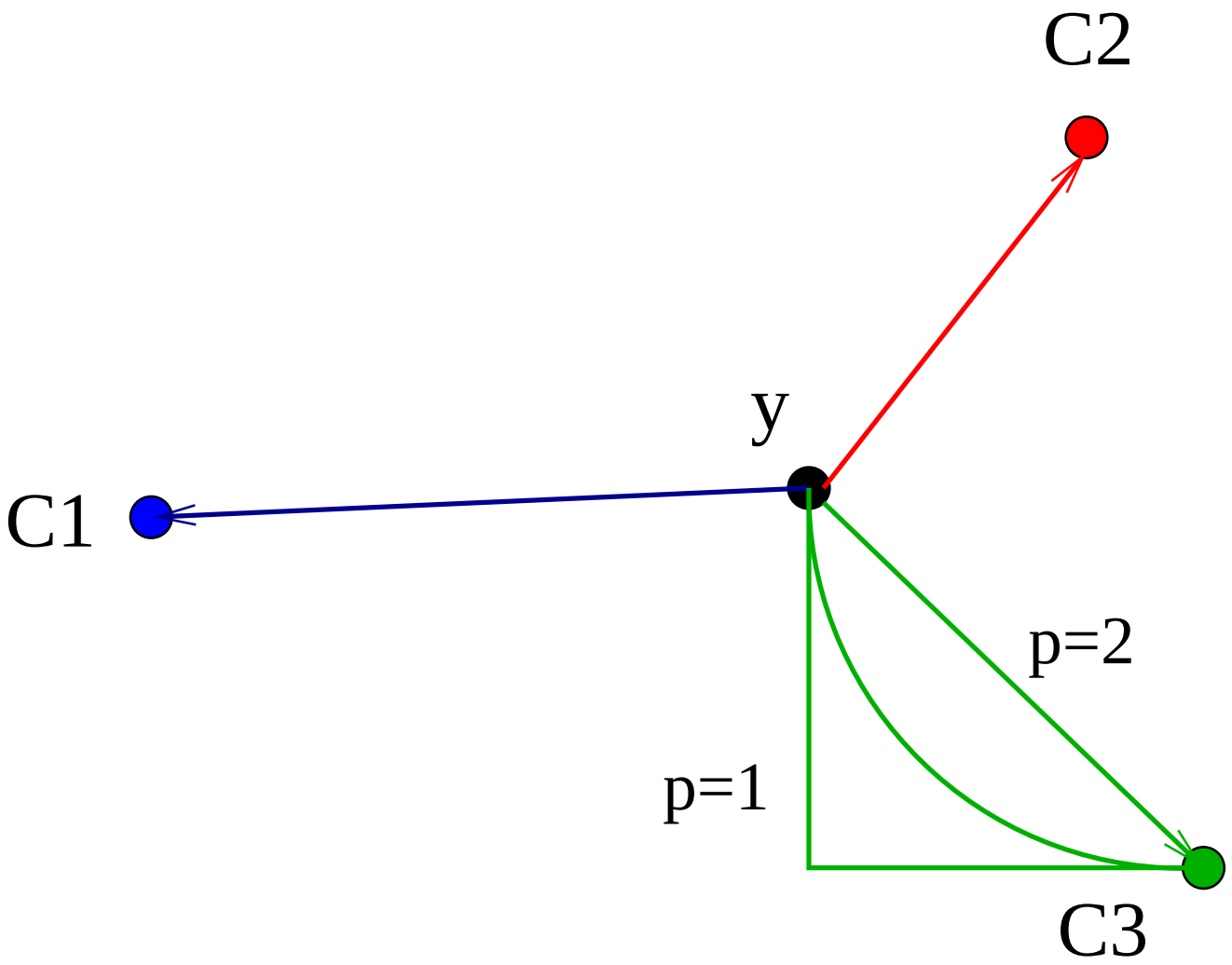
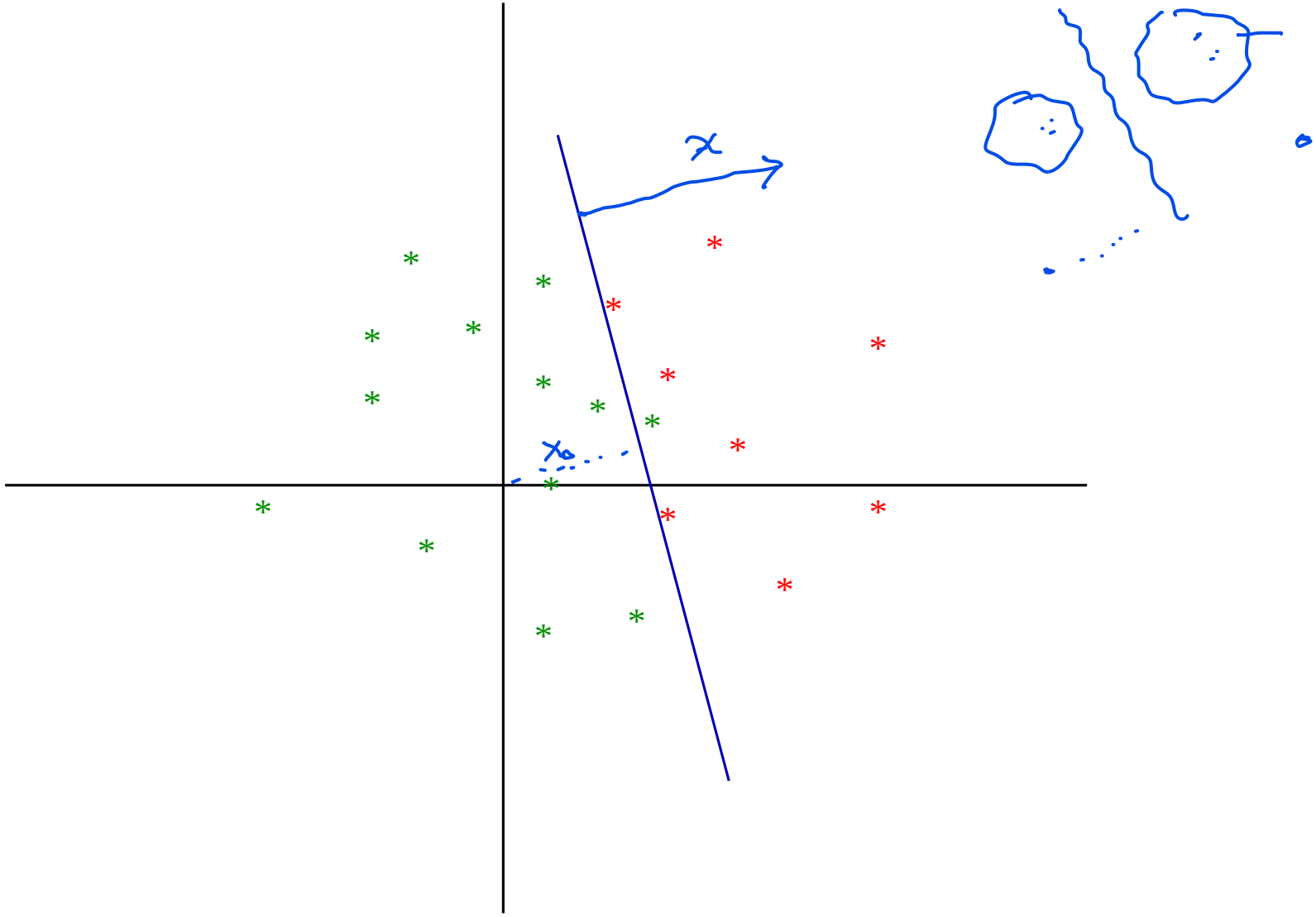


Figure 2: Facility Location at Point $\mathbf{y}$.

Linear Classifier: Support Vector Machine and Logistic Regression



Data Classification: Supporting Vector Machine I

A powerful binary-classification method is the Supporting Vector Machine (SVM).

Let the first class, say in Red, data points i be denoted by $\mathbf{a}_i \in R^d$, $i = 1, ..., n_1$ and the second class data points j be denoted by $\mathbf{b}_j \in R^d$, $j = 1, ..., n_2$. We like to find a hyperplane, slope vector $\mathbf{x}$ and intersect scalar x_0, to separate the two data classes:

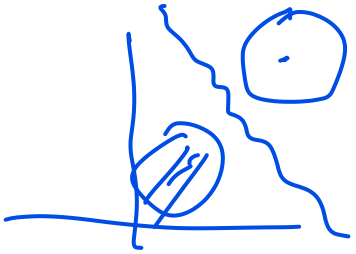
$$\begin{aligned} \text{subject to} \quad & \mathbf{a}_i^T \mathbf{x} + x_0 \geq 1, \ \forall i, \\ & \mathbf{b}_j^T \mathbf{x} + x_0 \leq -1, \ \forall j. \end{aligned}$$

This is a linear program with the null objective!

Data Classification: Supporting Vector Machine II

If strict separation is impossible, we then minimize error variable β

$$\begin{aligned} \text{minimize} \quad & \beta \\ \text{subject to} \quad & \mathbf{a}_i^T \mathbf{x} + x_0 + \beta \geq 1, \ \forall i, \\ & \mathbf{b}_j^T \mathbf{x} + x_0 - \beta \leq -1, \ \forall j, \\ & \beta \geq 0. \end{aligned}$$



Frequently we add the regularization term on the slope vector

$$\begin{aligned} \text{minimize} \quad & \beta + \mu \|\mathbf{x}\|^2 \\ \text{subject to} \quad & \mathbf{a}_i^T \mathbf{x} + x_0 + \beta \geq 1, \ \forall i, \\ & \mathbf{b}_j^T \mathbf{x} + x_0 - \beta \leq -1, \ \forall j, \\ & \beta \geq 0, \end{aligned}$$

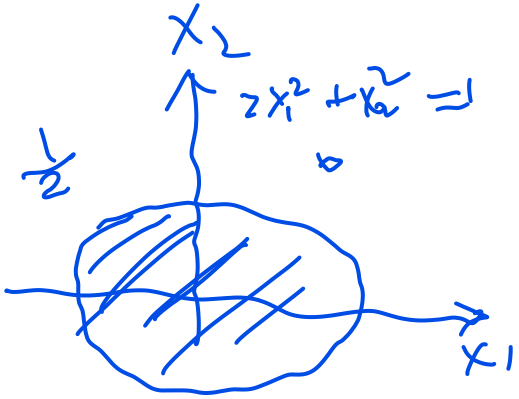
where μ is a fixed positive regularization parameter.

This becomes a constrained quadratic program (QP). If $\mu = 0$, then it is a linear program (LP)!

Supporting Vector Machine: Ellipsoidal Separation?

$$
\begin{aligned}
\text{minimize} \quad & \text{trace}(X) + \|\mathbf{x}\|^2 \\
\text{subject to} \quad & \mathbf{a}_i^T X \mathbf{a}_i + \mathbf{a}_i^T \mathbf{x} + x_0 \geq 1, \ \forall i, \\
& \mathbf{b}_j^T X \mathbf{b}_j + \mathbf{b}_j^T \mathbf{x} + x_0 \leq -1, \ \forall j, \\
& X \succeq \mathbf{0}.
\end{aligned}
$$

This type of problems is semidefinite programming (SDP). When the problem is not separable:



$$
\begin{aligned}
\text{minimize} \quad & \beta + \mu(\text{trace}(X) + \|\mathbf{x}\|^2) \\
\text{subject to} \quad & \mathbf{a}_i^T X \mathbf{a}_i + \mathbf{a}_i^T \mathbf{x} + x_0 + \beta \geq 1, \ \forall i, \\
& \mathbf{b}_j^T X \mathbf{b}_j + \mathbf{b}_j^T \mathbf{x} + x_0 - \beta \leq -1, \ \forall j, \\
& \beta \geq 0, \\
& X \succeq \mathbf{0}.
\end{aligned}
$$

This is a mixed linear and SDP program.

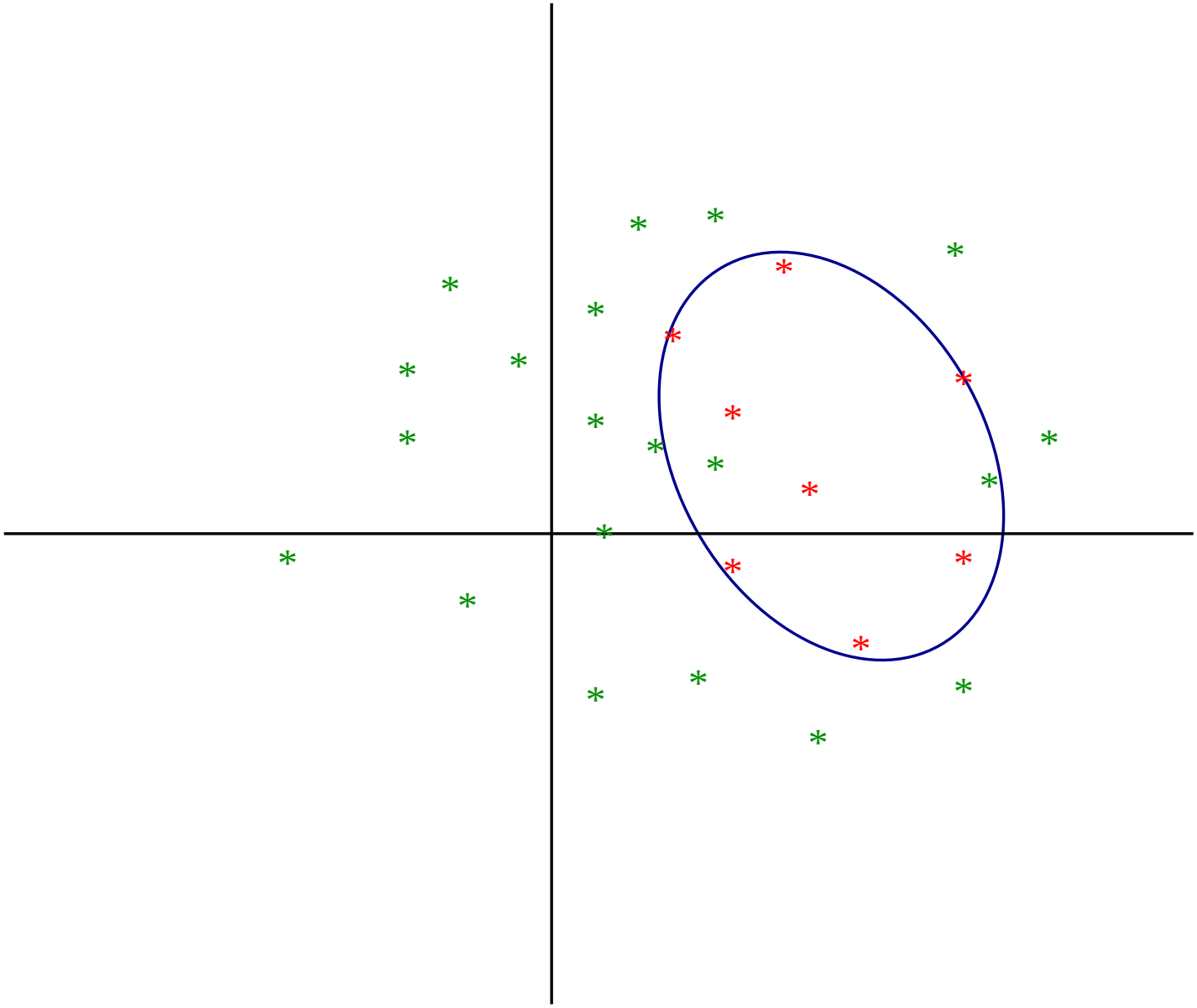


Figure 3: Quadratic Support Vector Machine