Mathematical Optimization Models and Applications

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

http://www.stanford.edu/˜yyye

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
- Linearly Constrained Optimization: If the constraint functions are linear/affine
- Conic Linear Optimization: If both the objective and the constraint functions are linear/affine, but variables in a convex cone.
- Quadratically Constrained Quadratic Optimization: If both the objective and the constraint functions are quadratic
- Nonlinear Optimization: If the constraints contain general nonlinear functions
- There are integer program, mixed-integer program etc.

Logistic Regression I

Given a data point $\mathbf{a}_i \in R^n$, according to the logistic model, the probability that it's in one class C is represented by

$$\frac{e^{\mathbf{a}_i^T\mathbf{x}+x_0}}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}.$$

Thus, for some training data points, we like to determine x_0 and $\mathbf{x} \in R^n$ such that

$$\frac{e^{\mathbf{a}_i^T\mathbf{x}+x_0}}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}} = \begin{cases} 1, & \text{if } \mathbf{a}_i \in C \\ 0, & \text{otherwise} \end{cases}.$$

Then the probability to give a "right answer" for all training data points is

$$\left(\prod_{\mathbf{a}_i \in C} \frac{e^{\mathbf{a}_i^T\mathbf{x}+x_0}}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}\right)\left(\prod_{\mathbf{a}_i \notin C} \frac{1}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}\right)$$

Logistic Regression II

Therefore, we like to maximize the probability

$$\left(\prod_{\mathbf{a}_i \in C} \frac{e^{\mathbf{a}_i^T\mathbf{x}+x_0}}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}\right)\left(\prod_{\mathbf{a}_i \notin C} \frac{1}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}\right) = \left(\prod_{\mathbf{a}_i \in C} \frac{1}{1+e^{-\mathbf{a}_i^T\mathbf{x}-x_0}}\right)\left(\prod_{\mathbf{a}_i \notin C} \frac{1}{1+e^{\mathbf{a}_i^T\mathbf{x}+x_0}}\right),$$

which is equivalently to maximize

$$-\left(\sum_{\mathbf{a}_i \in C} \ln(1+e^{-\mathbf{a}_i^T\mathbf{x}-x_0})\right) - \left(\sum_{\mathbf{a}_i \notin C} \ln(1+e^{\mathbf{a}_i^T\mathbf{x}+x_0})\right).$$

Or

$$\min_{x_0,\mathbf{x}} \quad \left(\sum_{\mathbf{a}_i \in C} \ln(1+e^{-\mathbf{a}_i^T\mathbf{x}-x_0})\right) + \left(\sum_{\mathbf{a}_i \notin C} \ln(1+e^{\mathbf{a}_i^T\mathbf{x}+x_0})\right).$$

This is an unconstrained optimization problem, where the objective is a convex function of decision variables.

Linear Programming

Given constraint matrix $A \in R^{m \times n}$, the objective coefficient vector $\mathbf{c} \in R^n$ and the right-hand-side vector $\mathbf{b} \in R^m$, we like

$$\begin{array}{rl} \text{min(or max)}_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x}\,\{\leq, =, \geq\}\,\mathbf{b}, \\ & \mathbf{x}\,\{\geq, \leq\}\,\mathbf{0}. \end{array}$$

- solution (decision, point): any specification of values for all decision variables, regardless of whether it is a desirable or even allowable choice; feasible solution: a solution for which all the constraints are satisfied; feasible region (constraint set, feasible set): the collection of all feasible solution; interior, boundary, extreme point (corner) or basic feasible solution.
- objective function contour (iso-profit, iso-cost line); optimal solution (optimum): a feasible solution that has the most favorable value of the objective function; optimal (objective) value: the value of the objective function evaluated at an optimal solution
- active (binding) constraint, inactive constraint, redundant constraint...

Linearly Constrained Programs in Standard Form

Linear Programming (LP)

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{array}$$

Basic Feasible Solution: Select m independent columns from A and solve for $\mathbf{x}_B$ such that $A_B\mathbf{x}_B = \mathbf{b}$ and set $\mathbf{x}_N = \mathbf{0}$. If $\mathbf{x}_B \geq \mathbf{0}$, then $\mathbf{x}_B$ together with $\mathbf{x}_N$ is called a BFS, and it is an extreme solution of the feasible region.

Linearly Constrained Optimization Problem (LCOP)

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & A\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{array}$$

LP and LCOP Examles: Sparsest Data Fitting

We want to find a sparsest solution to fit exact data measurements, that is, to minimize the number of non-zero entries in $\mathbf{x}$ such that $A\mathbf{x} = \mathbf{b}$:

$$\begin{array}{ll} \text{minimize} & \|\mathbf{x}\|_0 = |\{j : x_j \neq 0\}| \\ \text{subject to} & A\mathbf{x} = \mathbf{b}. \end{array}$$

Sometimes this objective can be accomplished by

$$\begin{array}{ll} \text{minimize} & \|\mathbf{x}\|_1 = \sum_{j=1}^{n} |x_j| \\ \text{subject to} & A\mathbf{x} = \mathbf{b}. \end{array}$$

This is a linear program!

Sparsest Data Fitting continued

It can be equivalently (?) represented by

$$\begin{array}{ll} \text{minimize} & \sum_{j=1}^{n} y_j \\ \text{subject to} & A\mathbf{x} = \mathbf{b},\ -\mathbf{y} \le \mathbf{x} \le \mathbf{y}; \end{array}$$

or

$$\begin{array}{ll} \text{minimize} & \sum_{j=1}^{n} (x'_j + x''_j) \\ \text{subject to} & A(\mathbf{x}' - \mathbf{x}'') = \mathbf{b},\ \mathbf{x}' \ge \mathbf{0},\ \mathbf{x}\text{"} \ge \mathbf{0}. \end{array}$$

Both are linear programs!

Sparsest Data Fitting continued

A better approximation of the objective can be accomplished by

$$\begin{array}{ll} \text{minimize} & \|\mathbf{x}\|_p := \left(\sum_{j=1}^n |x_j|^p\right)^{1/p} \\ \text{subject to} & A\mathbf{x} = \mathbf{b}; \end{array}$$

for some $0 < p < 1$. Or

$$\begin{array}{ll} \text{minimize} & \|\mathbf{x}\|_p^p = \sum_{j=1}^n |x_j|^p \\ \text{subject to} & A\mathbf{x} = \mathbf{b}. \end{array}$$

This is a linearly constrained optimization problem!

Conic LP and QCQP

Conic Linear Programming (CLP)

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

Second-Order Cone Program (SOCP): when K is a second-order cone

Semidefinite Cone Program (SDP): when K is a semidefinite matrix cone

Quadratically Constrained Quadratic Programming (QCQP)

$$\begin{array}{ll} \text{minimize} & q_0(\mathbf{x}) \\ \text{subject to} & q_i(\mathbf{x}) <=, \leq> 0, \ \forall i = 1, ..., m \end{array}$$

where

$$q_i(\mathbf{x}) = \mathbf{x}^T Q_i \mathbf{x} + \mathbf{c}_i^T \mathbf{x}.$$

CLP Example: Facility Location

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

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

Or equivalently (?)

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

This is a p-order conic linear program for $p \ge 1$. In particular, when $p = 2$, it is an SOCP problem.

For simplicity, consider $m = 3$.

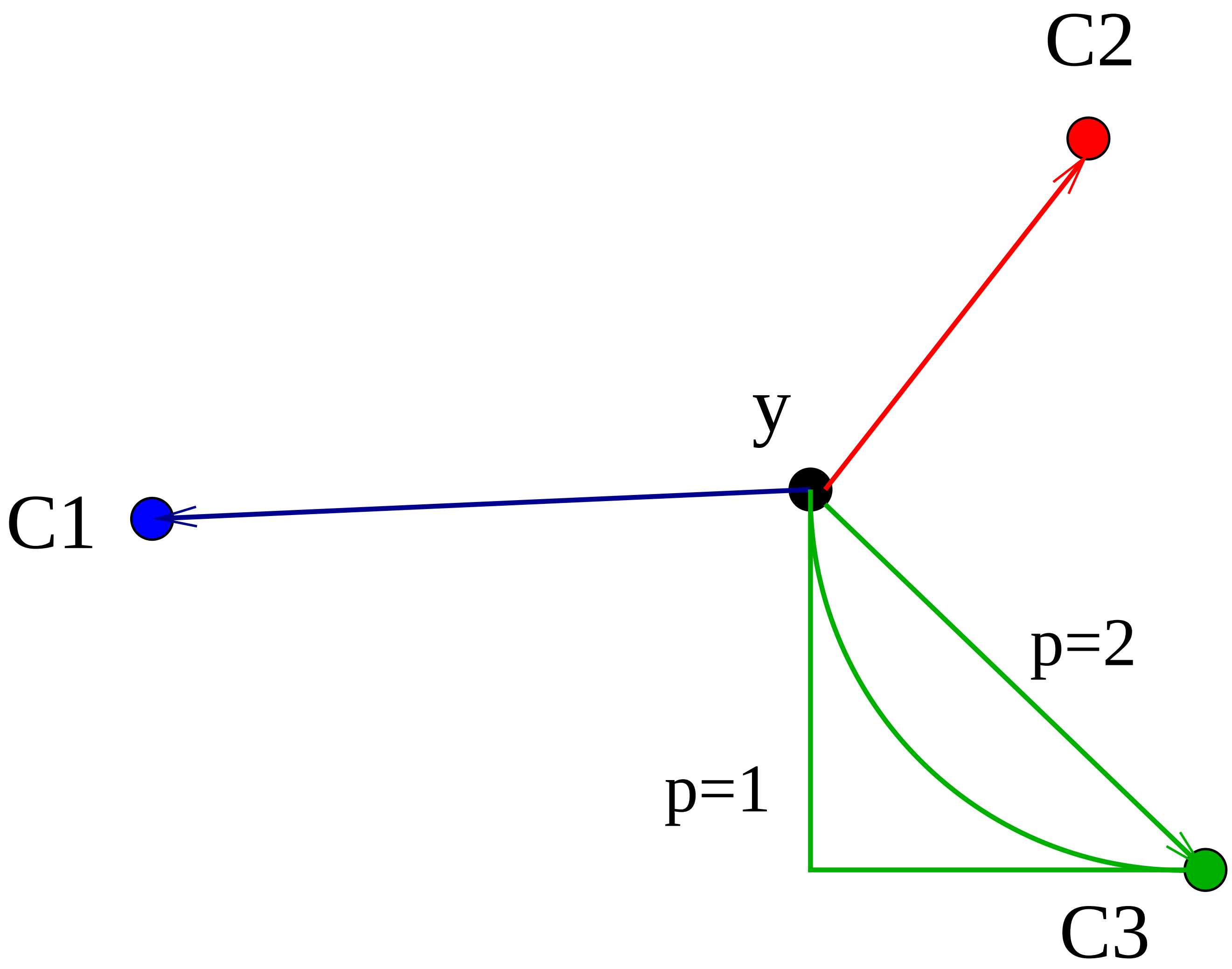

Figure 1: Facility Location at Point $\mathbf{y}$.

QCQP: Graph Realization and Sensor Network Localization

Given a graph $G = (V, E)$ and sets of non–negative weights, 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 vertices of G in $\mathbf{R}^d$ such that
- the Euclidean distance between a pair of adjacent vertices (i, j) equals to (or bounded by) the prescribed weight $d_{ij} \in E$.

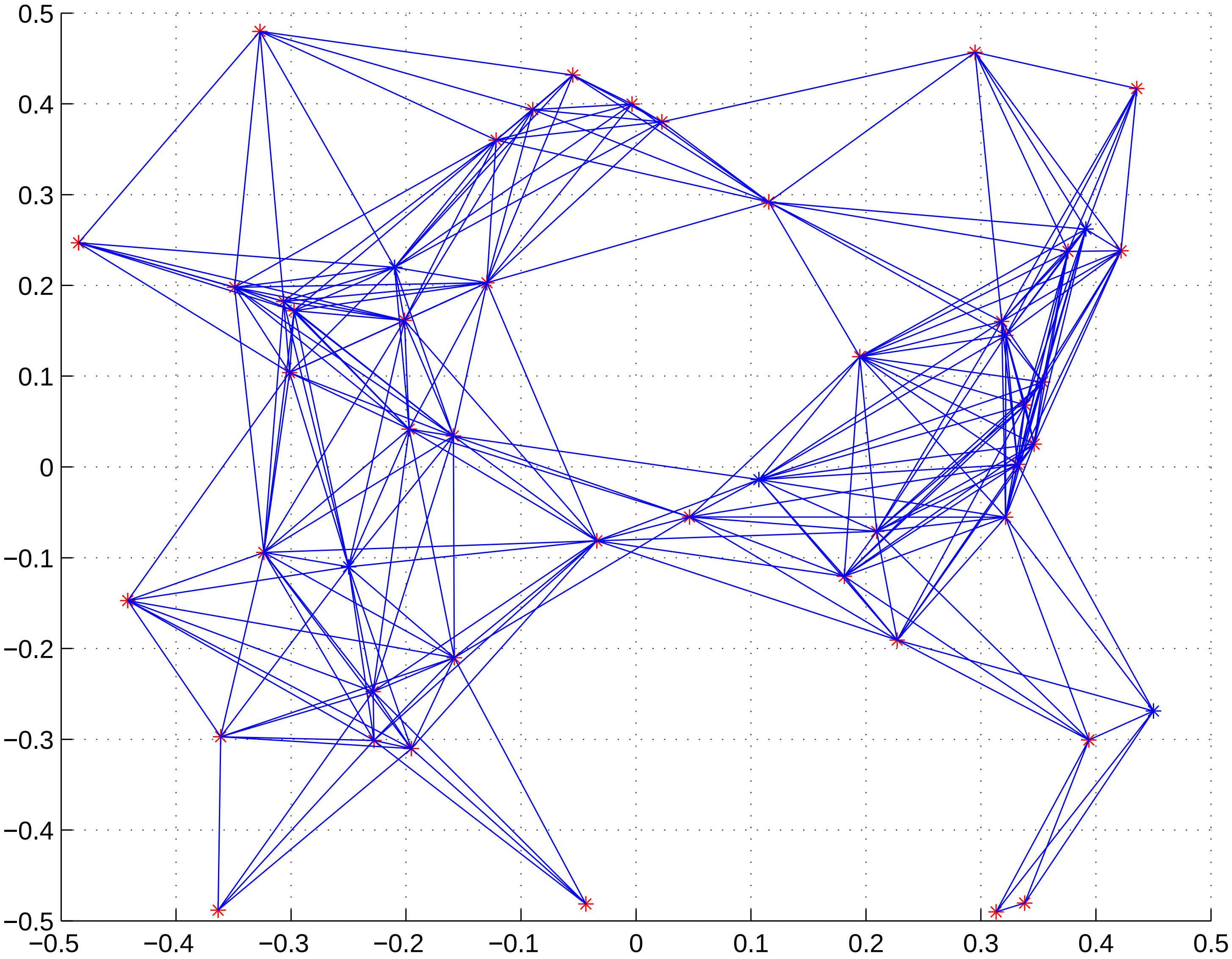

Figure 2: 50-node 2-D Sensor Localization.

Mathematical Sensor Localization Model

Given anchors $\mathbf{a}_k \in \mathbf{R}^d$, $d_{ij} \in N_x$, and $\hat{d}_{kj} \in N_a$, find $\mathbf{x}_i \in \mathbf{R}^d$ such that

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 = d_{ij}^2, \; \forall\, (i,j) \in N_x, \; i < j,$$
$$\|\mathbf{a}_k - \mathbf{x}_j\|^2 = \hat{d}_{kj}^2, \; \forall\, (k,j) \in N_a,$$

This is a QCQP, and it can be relaxed to SOCP or SDP.

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

Matrix Representation of SNL and SDP Relaxation

Let $X = [\mathbf{x}_1\ \mathbf{x}_2\ ...\ \mathbf{x}_n]$ be the $d \times n$ matrix that needs to be determined and $\mathbf{e}_j$ be the vector of all zero except 1 at the jth position. Then

$$\mathbf{x}_i - \mathbf{x}_j = X(\mathbf{e}_i - \mathbf{e}_j) \quad \text{and} \quad \mathbf{a}_k - \mathbf{x}_j = [I\ \ X](\mathbf{a}_k; -\mathbf{e}_j)$$

so that

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 = (\mathbf{e}_i - \mathbf{e}_j)^T X^T X(\mathbf{e}_i - \mathbf{e}_j)$$

$$\|\mathbf{a}_k - \mathbf{x}_j\|^2 = (\mathbf{a}_k; -\mathbf{e}_j)^T [I\ \ X]^T [I\ \ X](\mathbf{a}_k; -\mathbf{e}_j) =$$

$$(\mathbf{a}_k; -\mathbf{e}_j)^T \begin{pmatrix} I & X \\ X^T & X^T X \end{pmatrix} (\mathbf{a}_k; -\mathbf{e}_j).$$

Or, equivalently,

$$
\begin{aligned}
&(\mathbf{e}_i - \mathbf{e}_j)^T Y (\mathbf{e}_i - \mathbf{e}_j) = d_{ij}^2, \ \forall\, i, j \in N_x, \ i < j, \\
&(\mathbf{a}_k; -\mathbf{e}_j)^T \begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} (\mathbf{a}_k; -\mathbf{e}_j) = \hat{d}_{kj}^2, \ \forall\, k, j \in N_a, \\
&Y = X^T X.
\end{aligned}
$$

Relax $Y = X^T X$ to $Y \succeq X^T X$, which is equivalent to matrix inequality:

$$
\begin{pmatrix} I & X \\ X^T & Y \end{pmatrix} \succeq \mathbf{0}.
$$

This matrix has rank at least d; if it's d, then $Y = X^T X$, and the converse is also true.

The problem is now an SDP problem.

More LCOP and CLP Examples: Portfolio Management

For expected return vector $\mathbf{r}$ and co-variance matrix V of an investment portfolio, one management model is:

$$\begin{array}{ll} \text{minimize} & \mathbf{x}^T V \mathbf{x} \\ \text{subject to} & \mathbf{r}^T \mathbf{x} \geq \mu, \\ & \mathbf{e}^T \mathbf{x} = 1, \ \mathbf{x} \ \geq \ \mathbf{0}, \end{array}$$

where $\mathbf{e}$ is the vector of all ones.

This is a convex quadratic program, a special case of LCOP.

QCQP Examples: Robust Portfolio Management

In applications, $\mathbf{r}$ and V may be estimated under various scenarios, say $\mathbf{r}_i$ and V_i for $i = 1, ..., m$. Then, we like

$$\begin{array}{rl} \text{minimize} & \max_i \mathbf{x}^T V_i \mathbf{x} \\ \text{subject to} & \min_i \mathbf{r}_i^T \mathbf{x} \geq \mu, \\ & \mathbf{e}^T \mathbf{x} = 1, \ \mathbf{x} \ \geq \ \mathbf{0}. \end{array}$$

$$\begin{array}{rl} \text{minimize} & \alpha \\ \text{subject to} & \mathbf{r}_i^T \mathbf{x} \geq \mu, \ \forall i \\ & \mathbf{x}^T V_i \mathbf{x} \leq \alpha, \ \forall i \\ & \mathbf{e}^T \mathbf{x} = 1, \ \mathbf{x} \ \geq \ \mathbf{0}. \end{array}$$

This is a quadratically constrained program.

Recall Data Classification: Supporting Vector Machine

Suppose we have two-class discrimination data. We assign the first class with 1 and the second with -1 for a binary varible. A powerful discrimination method is the Supporting Vector Machine (SVM).

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

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

This is a quadratic program.

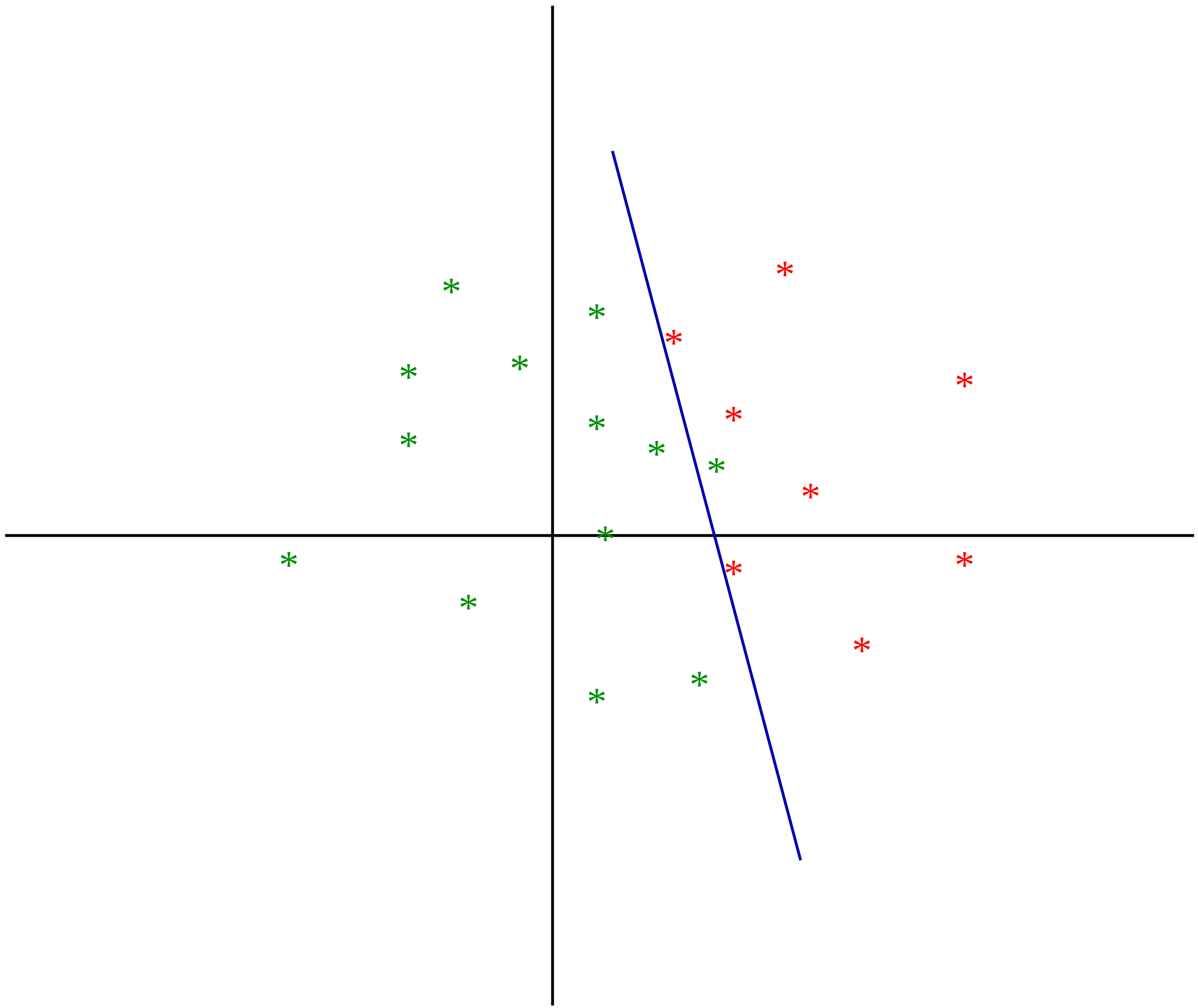

Figure 3: Linear Support Vector Machine

Supporting Vector Machine: Ellipsoidal Separation?

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

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

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

This problems is a mixed linear and SDP program.

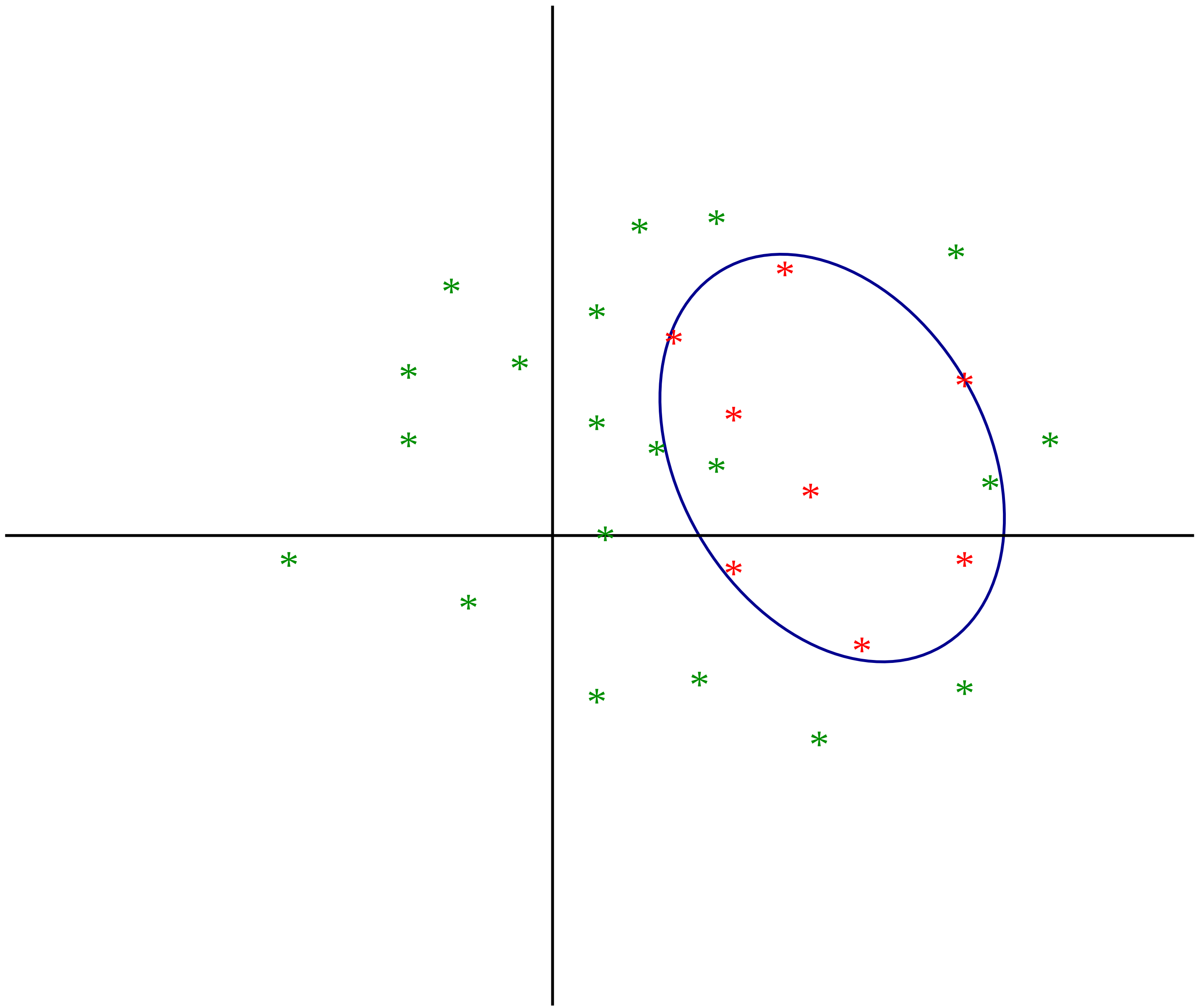

Figure 4: Quadratic Support Vector Machine