Nonlinear Optimization: The art of modeling

INSEAD, Spring 2006

Jean-Philippe Vert

Ecole des Mines de Paris

Jean-Philippe.Vert@mines.org

The art of modeling

Objective: to distill the real-world as accurately and succinctly as possible into a *quantitative model*

- Dont want models to be *too generalized*: might not draw much real world value from your results.

  Ex: Analyzing traffic flows assuming every person has the same characteristics.

- Dont want models to be *too specific*: might lose the ability to solve problems or gain insights.
  Ex: Trying to analyze traffic flows by modeling every single individual using different assumptions.

The four-step rule for modeling

- Sort out *data and parameters* from the verbal description
- Define the set of *decision variables*
- Formulate the *objective function* of data and decision variables
- Set up equality and/or inequality *constraints*

Problem reformulation

- Only few problems can be solved *efficiently* (LP, QP, ...)
- Your problem can often be *reformulated* in an (almost) equivalent problem that can be solved, up to:
  - adding/removing variables
  - adding/removing constraints
  - modifying the objective function

Problem reformulation is key for practical optimization!

Model 1: a cheap and healthy diet

A healthy diet contains m different nutrients in quantities at least equal to $b_1, \ldots, b_m$. We can compose such a diet with n different food. The j's food has a cost c_j, and contains an amount a_{ij} of nutrients i ($i = 1, \ldots, m$).

How to determine the cheapest healthy diet that satisfies the nutritional requirements?

A cheap and healthy diet (cont.)

- *Decision variables*: the quantities of the n different food (nonnegative scalars)
- *Objective function*: the cost of the diet, to be minimized.
- *Constraints*: be healthy, i.e., lower bound on the quantities of each food.

A cheap and healthy diet (cont.)

Let $x_1, \ldots, x_n$ the quantities of the n different food. The problem can be formulated as the *LP*:

$$\begin{aligned} \textit{minimize} \quad & \sum_{j=1}^{n} x_j c_j \\ \textit{subject to} \quad & \sum_{j=1}^{n} x_j a_{ij} \geq b_i \,, \quad i = 1, \ldots, m \,, \\ & x_j \geq 0 \,, \quad j = 1, \ldots, n \,. \end{aligned}$$

This is easily solved (see “Linear Programming” course)

Model 2: Air traffic control

Air plane $j, j = 1, \ldots, n$ arrives at the airport within the time interval $[a_j, b_j]$ in the order of $1, 2, \ldots, n$. The airport wants to find the arrival time for each air plane such that the narrowest metering time (inter-arrival time between two consecutive airplanes) is the greatest.

Air traffic control (cont.)

- *Decision variables*: the arrival times of the planes.
- *Objective function*: the narrowest metering time, to be maximized.
- *Constraints*: arrive in the good order, and in the good time slots.

Air traffic control (cont.)

Let t_j be the arrival time of plane j . Then optimization problem translates as:

$$\begin{array}{ll} \text{maximize} & \min\limits_{j=1,\ldots,n-1} (t_{j+1} - t_j) \\ \text{subject to} & a_j \leq t_j \leq b_j \,, \quad j = 1, \ldots, n \,, \\ & t_j \leq t_{j+1} \,, \quad j = 1, \ldots, n-1 \,. \end{array}$$

In order to solve it we need to *reformulate* it in a simpler way.

Air traffic control (cont.)

Reformulation with a *slack variable*:

$$
\begin{aligned}
\text{maximize} \quad & \Delta \\
\text{subject to} \quad & a_j \leq t_j \leq b_j\,, \quad j = 1, \ldots, n\,, \\
& t_j \leq t_{j+1}\,, \quad j = 1, \ldots, n-1\,, \\
& \Delta \leq \min_{j=1,\ldots,n-1} \left(t_{j+1} - t_j\right)\,.
\end{aligned}
$$

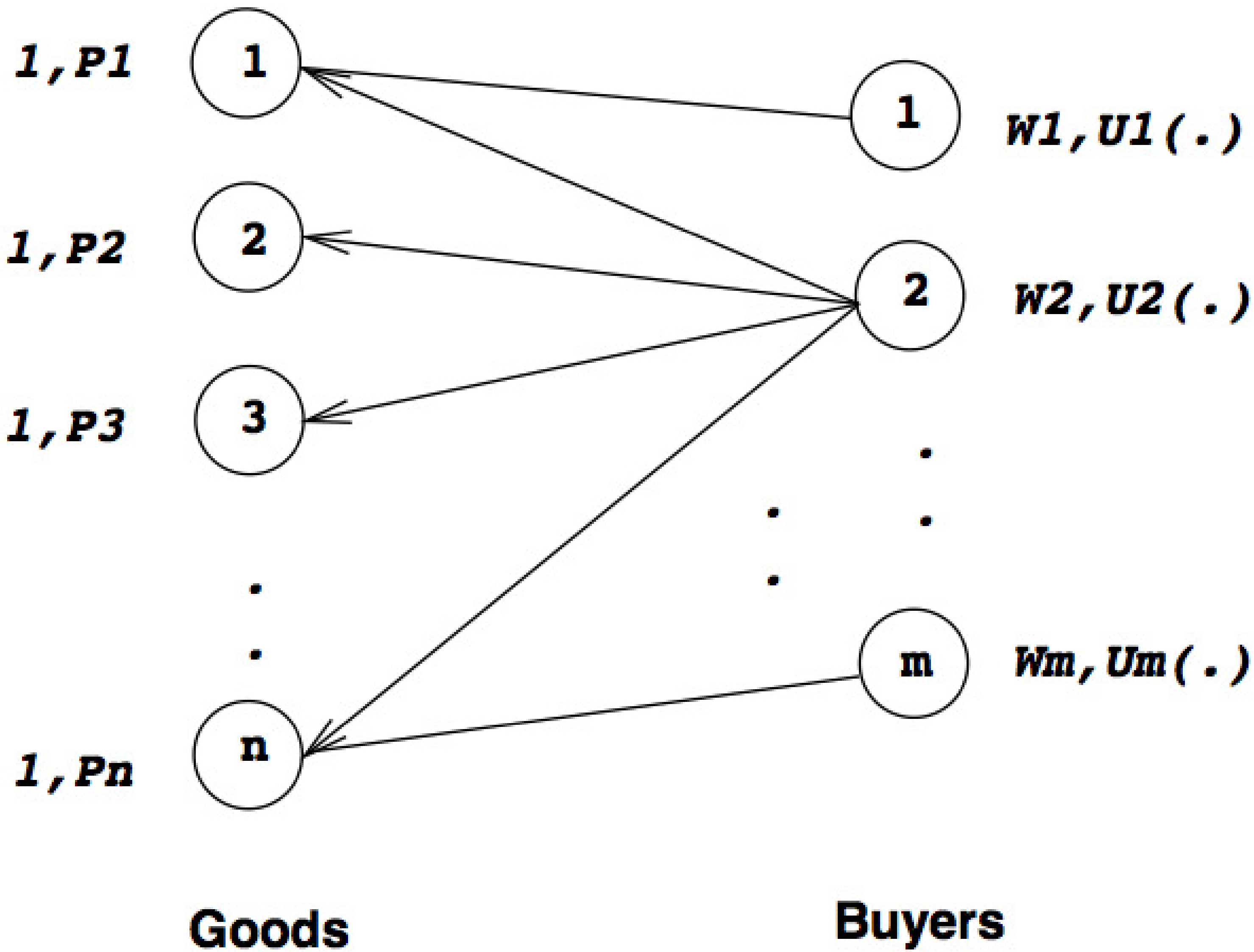
Equivalent to the *LP* (and therefore easily solved):

$$
\begin{aligned}
\textit{maximize} \quad & \Delta \\
\textit{subject to} \quad & a_j \leq t_j \leq b_j\,, \quad j = 1, \ldots, n\,, \\
& t_j \leq t_{j+1}\,, \quad j = 1, \ldots, n-1\,, \\
& \Delta \leq t_{j+1} - t_j\,, \quad j = 1, \ldots, n-1\,.
\end{aligned}
$$

Model 3: Fisher's exchange market

Buyers have money (w_i) to buy *goods* and maximize their individual *utility functions*; *Producers* sell their goods for money. The *equilibrium price* is an assignment of prices to goods so as when every buyer buys an maximal bundle of goods then the market clears, meaning that all the money is spent and all goods are sold.

Fisher's exchange market



Buyer's strategies

Let $x_{i,j}$ the amount of good $j \in G$ bought by buyer $i \in B$. Let $U_i(x) = U_i(x_{i,1}, \ldots, x_{i,G})$ be the utility function of buyer $i \in B$.

Buyer $i \in B$'s optimization problem for given prices $p_j, j \in G$ is the following LP:

$$\begin{aligned} \textit{maximize} \quad & U_i(x) \\ \textit{subject to} \quad & \sum_{j \in G} p_j x_{ij} \leq w_i \,, \\ & x_{ij} \geq 0 \,, \quad \forall j \in G \,. \end{aligned}$$

Depending on U this is a LP (linear), QP (quadratic), LCCP (convex)...

Equilibrium price

Without losing generality, assume that the amount of each good is 1. The equilibrium price vector p^* is the one that ensures:

$$\sum_{i \in B} x^*(p^*)_{ij} = 1$$

for all goods $j \in G$, where $x^*(p)$ are the optimal bundle solutions.

Example of Fisher's market

Buyer $1, 2$'s optimization problems for given prices p_x, p_y assuming linear utility functions:

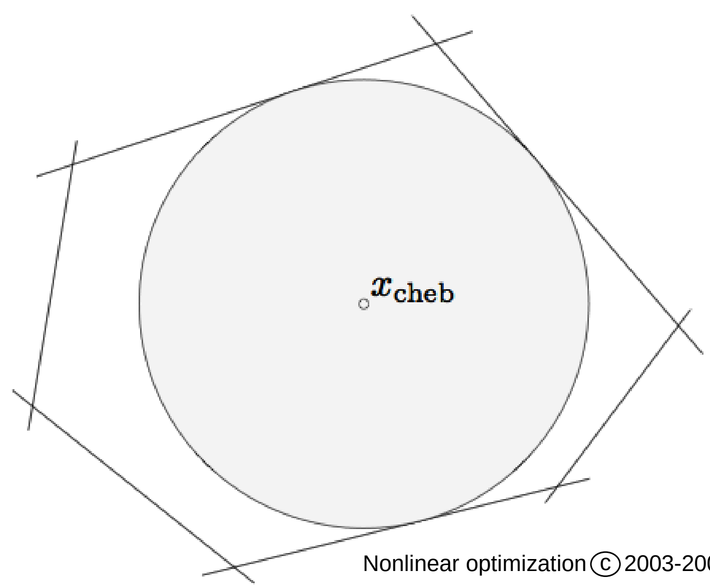
$$\begin{aligned}
\text{maximize} \quad & 2x_1 + y_1 \\
\text{subject to} \quad & p_x x_1 + p_y y_1 \leq 5\,, \\
& x_1, y_1 \geq 0\,; \\
\text{maximize} \quad & 3x_2 + y_2 \\
\text{subject to} \quad & p_x x_2 + p_y y_2 \leq 8\,, \\
& x_2, y_2 \geq 0\,.
\end{aligned}$$

Model 4: Chebyshev center

How to find the largest Euclidean ball that lies in a polyhedron described by a set of linear inequalities:

$$\mathcal{P} = \left(x \in \mathbb{R}^n \mid a_i^\top x \leq b_i, i = 1, \ldots, m \right) .$$

The center of the optimal ball is called the *Chebyshev center of the polyhedron*; it is the point deepest inside the polyhedron, i.e., farthest from the boundary.



Chebyshev center (cont.)

The variables are the center $x_c \in \mathbb{R}^n$ and the radius $r \geq 0$ of the ball:

$$\mathcal{B} = (x_c + u \mid \| u \|_2 \leq r) \; .$$

The problem is then

$$\begin{aligned} \text{maximize} \quad & r \\ \text{subject to} \quad & \mathcal{B} \subseteq \mathcal{P} \, . \end{aligned}$$

We now need to translate the constraint into equations.

Chebyshev center (cont.)

For a single half-space defined by the equation $a_i^\top x \leq b_i, \mathcal{B}$ is on the correct halfspace iff it holds that:

$$\| u \|_2 \leq r \implies a_i^\top (x_c + u) \leq b_i \,.$$

But the maximum value that $a_i^\top u$ takes when $\| u \|_2 \leq r$ is $r \| a_i \|_2$. Therefore the constraint for a single half-space can be rewritten as:

$$a_i^\top x_c + r \| a_i \|_2 \leq b_i \,.$$

Chebyshev center (cont.)

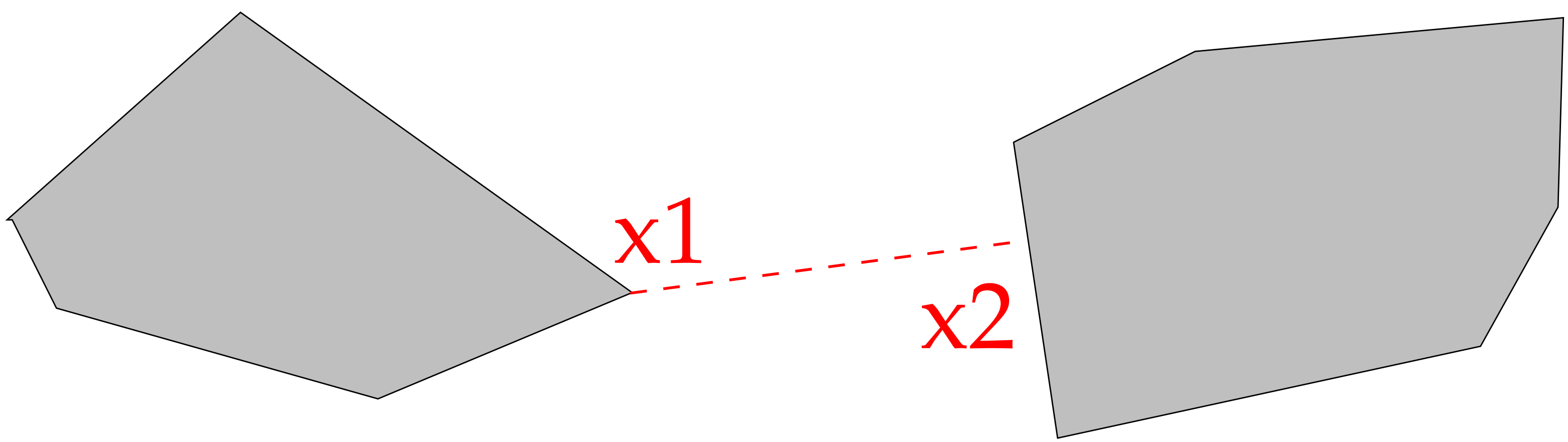
The Chebyshev center is therefore found by solving the following *LP*:

$$\begin{array}{ll} \textit{maximize} & r \\ \textit{subject to} & a_i^\top x_c + r \| a_i \|_2 \leq b_i \,, \quad i = 1, \ldots, m \,. \end{array}$$

Model 5: Distance between polyhedra

How to find the distance between two polyhedra $\mathcal{P}_1$ and $\mathcal{P}_2$ defined by two sets of linear inequalities:

$$\mathcal{P}_1 = \left(x \in \mathbb{R}^n \mid A_1 x \leq b_1\right) ,$$
$$\mathcal{P}_2 = \left(x \in \mathbb{R}^n \mid A_2 x \leq b_2\right) .$$



Distance between polyhedra (cont.)

The distance between two sets can be written as a minimum:

$$d(\mathcal{P}_1, \mathcal{P}_2) = \min_{x_1 \in \mathcal{P}_1, x_2 \in \mathcal{P}_2} \| x_1 - x_2 \|_2 \, .$$

The squared distance is therefore the solution of the following *QP*:

$$\begin{aligned} \textit{minimize} \quad & \| x_1 - x_2 \|_2^2 \\ \textit{subject to} \quad & A_1 x_1 \leq b_1 \, , \\ & A_2 x_2 \leq b_2 \, . \end{aligned}$$

Model 6: Portfolio optimization

We consider a classical portfolio problem with n assets or stocks held over a period of time. The vector of relative price changes over an investment period $p \in \mathbb{R}^n$ is assumed to be random variable with known mean $\bar{p}$ and covariance Σ. We want to define an investment strategies, which minimizes the risk (variance) of the return, while ensuring an expected return above a threshold r_{min}.

This investment strategy has been proposed first by Markowitz*.*

Portfolio optimization (cont.)

The decision variable is the *portfolio* vector $x \in \mathbb{R}^n$, i.e., the amount of each asset x_i to buy, in dollars ($i = 1 \ldots, n$). We call B the total amount of dollars we can invest.
The *return* in dollars is $r = p^\top x$, where p is the vector of relative prices changes over the period. The return is therefore a random variable with mean and variance:

$$E(r) = \bar{p}^\top x \,,$$

$$Var(r) = x^\top \Sigma x \,.$$

Portfolio optimization (cont.)

The Markowitz portfolio optimization problem is therefore the following *QP*:

$$
\begin{aligned}
\textit{minimize} \quad & x^\top \Sigma x \\
\textit{subject to} \quad & \bar{p}^\top x \geq r_{min}\,, \\
& \sum_{i=1}^{n} x_i \leq B\,, \\
& x_i \geq 0\,, \quad i = 1, \ldots, n\,.
\end{aligned}
$$

Model 6: Predicting traffic accidents

We monitor everyday the number of traffic accidents in Paris, together with several other explanatory variables. The goal is to make a model to predict the number of accidents from the explanatory variables, by fitting a Poisson distribution with mean depending linearly on the explanatory variables by maximum likelihood on the historical data.

Traffic accidents (cont.)

The *Poisson distribution* is commonly used to model nonnegative integer-valued random variables Y (photon arrivals, traffic accidents...). It is defined by:

$$P(Y = k) = \frac{e^{-\mu}\mu^k}{k!},$$

where μ is the mean.
Here we assume that the number of accidents follows a Poisson distribution with a mean μ that depends linearly on the vector $x \in \mathbb{R}^n$ of explanatory variables:

$$\mu = a^\top x + b.$$

The parameters $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are called the *model parameters*, and must be set according to some principle.

Traffic accidents (cont.)

We are given a set of historical data that consists of pairs $(x_i, y_i), i = 1, \ldots, m$ where y_i is the number of traffic accidents and x_i is the vector of explanatory variables at day i. The *likelihood* of the parameters (a, b) is defined by:

$$
\begin{aligned}
l(a,b) &= \prod_{i=1}^{m} P(y_i \mid x_i) \\
&= \prod_{i=1}^{m} \frac{\left(a^\top x_i + b\right)^{y_i} \exp\left(-\left(a^\top x_i + b\right)\right)}{y_i!} .
\end{aligned}
$$

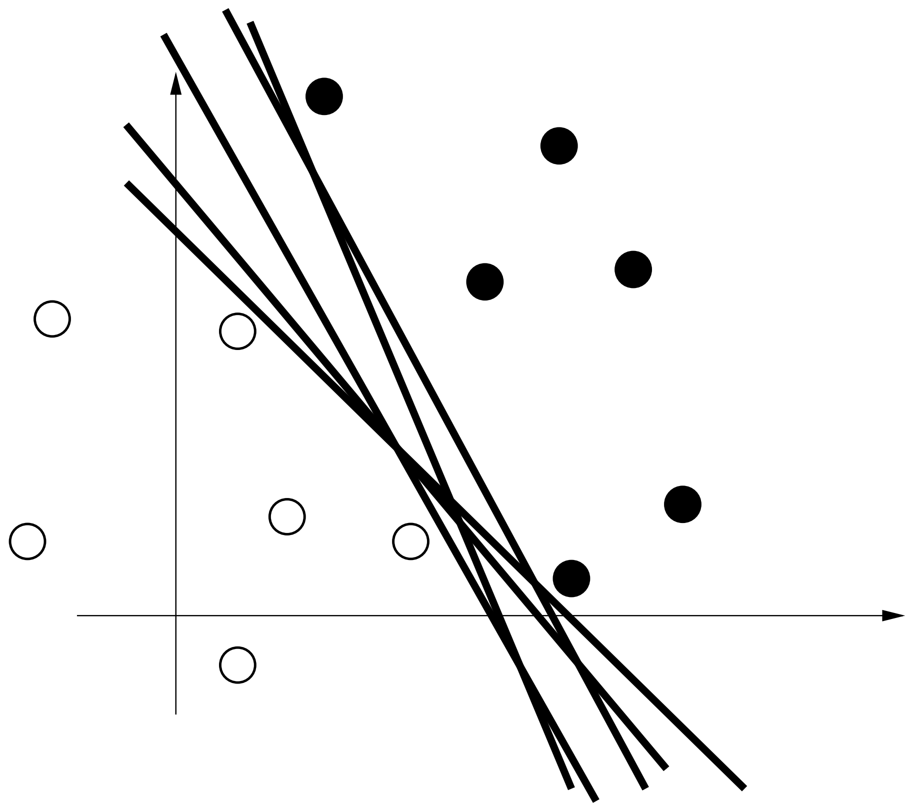
Traffic accidents (cont.)

Finding the parameter (a, b) by *maximum likelihood* is therefore obtained by solving the following *unconstrained convex problem*:

$$\textit{maximize} \quad \sum_{i=1}^{m} \left\{ y_i \log\left(a^\top x_i + b\right) - \left(a^\top x_i + b\right) \right\} .$$

Model 7: Robust linear discrimination

Given n points in $\mathbb{R}^p$ from two classes that can be linearly separated, find the linear separator that is the furthest away from the closest point.



Robust linear discrimination (cont.)

A linear hyperplane is defined by the equation:

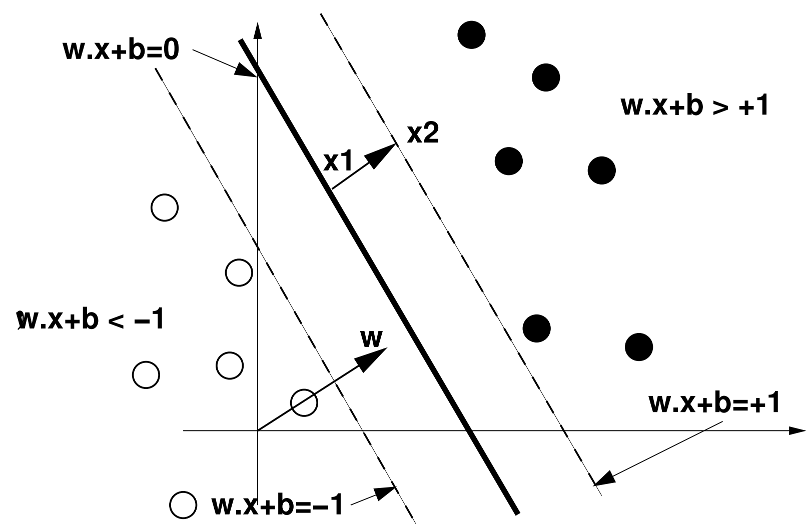
$$\mathcal{H}_0 = \left\{ x \in \mathbb{R}^p \ : \ a^\top x + b = 0 \right\} ,$$

for some $a \in \mathbb{R}^p$ and $b \in \mathbb{R}$.

Two parallel hyperplanes on either side are defined by:

$$\mathcal{H}_{-1} = \left\{ x \in \mathbb{R}^p \ : \ a^\top x + b = -1 \right\}$$

$$\mathcal{H}_{1} = \left\{ x \in \mathbb{R}^p \ : \ a^\top x + b = 1 \right\} .$$



Robust linear discrimination (cont.)

- The distance between $\mathcal{H}_{-1}$ and $\mathcal{H}_1$ is equal to $2/\| a \|_2$. Maximizing the distance is equivalent to *minimizing* $\| a \|_2$.
- Let $y_i \in \{-1, +1\}$ be the label of the point x_i. The point is on the correct region of the space iff:

$$\begin{cases} a^\top x_i + b \geq 1 & \text{if } y_i = 1 \,, \\ a^\top x_i + b \leq -1 & \text{if } y_i = -1 \,, \end{cases}$$

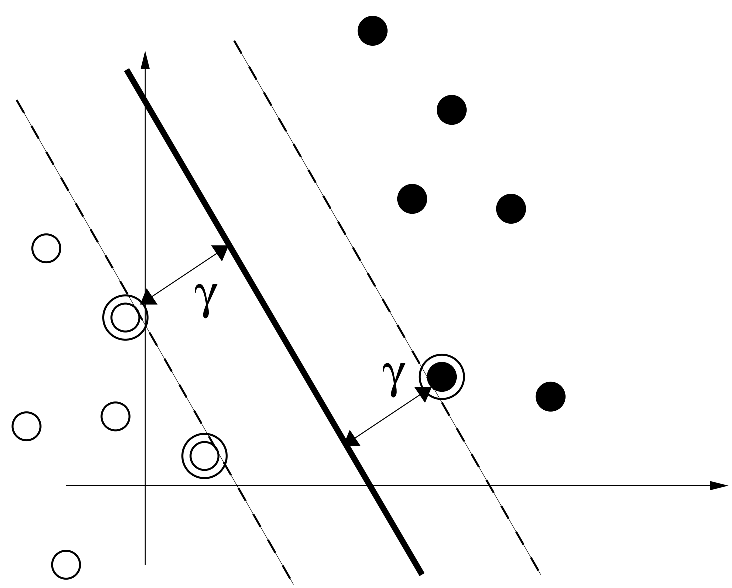
This is equivalent to:

$$y_i \left(a^\top x_i + b \right) \geq 1 \,.$$

Robust linear discrimination (cont.)

The optimal separating hyperplane is therefore the solution of the following *QP*:

$$\begin{array}{rl} \text{minimize} & \| a \|^2 \\ \text{subject to} & y_i \left(a^\top x_i + b \right) \geq 1 \, , i = 1, \ldots, n \, . \end{array}$$



Summary

- There are a few *general rules* to follow to transform a real-world problem into an optimization problem
- Most optimization problems are difficult to solve, therefore problem *reformulation* is often crucial for later practical optimization
- Problem formulation and reformulation involve a few classical *tricks* (e.g., slack variables) and much *experience and know-how* about which problems can efficiently be solved.