1 Sphere Enclosure

For $i = 1, \dots, m$, let B_i be a ball in $\mathbb{R}^n$ with center x_i, and radius $\rho_i \geq 0$. We wish to find a ball B of minimum radius that contains all the B_i for $i = 1, \dots, m$. Cast this problem as an SOCP.

2 Dual Norms and SOCP

Consider the problem

$$p^* := \min_{x \in \mathbb{R}^n} \|Ax - y\|_1 + \mu \|x\|_2, \tag{1}$$

where $A \in \mathbb{R}^{m \times n}$, $y \in \mathbb{R}^m$, and $\mu > 0$.

a) Express this (primal) problem in standard SOCP form.

b) Find the conic dual of the primal SOCP and express it as an SOCP.

   Hint: Recall that for every vector z, the following dual norm equalities hold:

   $$\|z\|_2 = \max_{u \,:\, \|u\|_2 \le 1} u^\top z, \quad \|z\|_1 = \max_{u \,:\, \|u\|_\infty \le 1} u^\top z.$$

c) Assume strong duality holds and that $m = 100$ and $n = 10^6$, i.e., A is 100×10^6. Which problem would you choose to solve using a numerical solver: the primal or the dual? Justify your answer.

   Remark (optional): Section 10.1.3 of the textbook of Calafiore and El Ghaoui consider a primal SOCP in the form:

   $$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \;\; i = 1, \ldots, m, \end{aligned} \tag{2}$$

   and shows that the conic dual of this SOCP can be written in the form:

   $$\begin{aligned} \max_{u,\lambda} \quad & \sum_{i=1}^m (u_i^T b_i - \lambda_i d_i) \\ \text{s.t.} \quad & \sum_{i=1}^m (A_i^T u_i - \lambda_i c_i) = -c, \\ & \|u_i\|_2 \le \lambda_i, \;\; i = 1, \ldots, m, \end{aligned} \tag{3}$$

   which is also an SOCP (note that the equality constraints for the dual in the textbook are wrongly written). The process by which we got from our primal problem in the form of (**??**) to the dual problem in (**??**), i.e. the conic dual of the primal, is the same as that in the textbook.

   In fact the conic dual of the primal problem in SOCP form is equivalent to the traditional Lagrangian dual of a problem that is equivalent to the primal. To see this, note that we can write the primal SOCP in (2) in the equivalent form:

   $$\begin{aligned} \min_{x,y,t} \quad & c^T x \\ \text{s.t.} \quad & \|y_i\|_2 \le t_i, \;\; i = 1, \ldots, m, \\ & y_i = A_i x + b_i, \;\; i = 1, \ldots, m, \\ & t_i = c_i^T x + d_i, \;\; i = 1, \ldots, m. \end{aligned} \tag{4}$$

   We can then write the Lagrangian for this problem, which has primal variables (x, y, t), using dual variables $(\lambda, \nu_1, \ldots, \nu_m, \mu)$, where $\lambda \in \mathbb{R}^m$ is the vector of dual variables for the first m inequality constraints, $\nu_i \in \mathbb{R}^{k_i}$ is the vector of dual variables for the i-th of the first set of m vector equality constraints, $1 \leq i \leq m$, and $\mu \in \mathbb{R}^m$ is the vector of dual variables for the last set of equality constraints. We can then go through the usual process of finding the dual objective function, which will turn out to be

$$g(\lambda, \nu_1, \ldots, \nu_m, \mu) = \begin{cases} -\sum_{i=1}^m (\nu_i^T b_i + \mu_i d_i) & \text{if } \sum_{i=1}^m (A_i^T \nu_i + \mu_i c_i) = c,\ \|\nu_i\|_2 \leq \lambda_i, \text{ and } \mu = \lambda, \\ -\infty & \text{otherwise.} \end{cases}$$

From this it will immediately follow that the resulting dual problem is equivalent to the conic dual in (3) by substituting for ν_i with $-u_i$ and eliminating μ (i.e. replacing it with λ).

If we go through this process in the case of the primal in SOCP form in (**??**), we will see that the resulting form of the primal satsifies Slater's condition. This explains why strong duality holds.

This is of course a rather elaborate and painful way to show strong duality. One can show that strong duality holds much more easily by appealing to a conic version of Slater's condition (see pg. 265 of the textbook of Boyd and Vandenberghe), which is beyond the scope of this class.

3 Robust machine learning

We consider a binary classification problem, where the prediction label associated with a test point $x \in \mathbb{R}^n$ is the form $\hat{y}(x) = \text{sign}(w^T x + v)$, with $(w, v) \in \mathbb{R}^n \times \mathbb{R}$ being the classifier weights. Given a training set X, y, with $X = [x_1, \ldots, x_m] \in \mathbb{R}^{n \times m}$ the data matrix, with data points $x_i \in \mathbb{R}^n$, $i = 1, \ldots, m$, and $y \in \{-1, 1\}^m$ the vector of corresponding labels, the training problem is to minimize the so-called hinge loss function:

$$\min_{w,v} \frac{1}{m} \sum_{i=1}^{m} \max(0, 1 - y_i(x_i^T w + v)). \tag{5}$$

We seek to find a classifier (w, v) that can be implemented with low precision (say, as an integer vector). To this end, we modify the training problem so that it accounts for the implementation error, when approximating the original optimal (full precision) weight vector w_* with a low-precision one, $\tilde{w}$. We bound the corresponding error as $\|\tilde{w} - w_*\|_\infty \leq \epsilon$ for some given absolute error bound $\epsilon > 0$; for example, if $\tilde{w}$ is the nearest integer vector, the error is bounded by $\epsilon = 0.5$. Then, we seek to solve the *robust counterpart* to (5):

$$\min_{w,v} \max_{\tilde{w} \,:\, \|\tilde{w}-w\|_\infty \leq \epsilon} \frac{1}{m} \sum_{i=1}^{m} \max(0, 1 - y_i(x_i^T \tilde{w} + v)). \tag{6}$$

(a) Justify the use of the hinge loss function in problem (5); in particular, explain geometrically what it means to have a zero loss.

(b) Show that without loss of generality, we can set $v = 0$, which we will do henceforth.

Hint: Think about adding a dimension to the data.

(c) Explain how to obtain a low-precision classifier once problem (6) is solved. What guarantees do we have on the training error?

(d) Show that the optimal value of problem (6) is bounded above by

$$\min_{w} \frac{1}{m} \sum_{i=1}^{m} \max(0, 1 - y_i(x_i^T w) + \epsilon \|x_i\|_1). \tag{7}$$

Hint: Solve the problem $\max_\delta \{\delta^T z \ : \ \|\delta\|_\infty \leq \epsilon\}$ first.

(e) Assume that the data set is normalized, in the sense that $\|x_i\|_1 = 1$, $i = 1, \ldots, m$. How would you solve problem (7) if you had code to solve (5) only?