Circular Separator

We need to check existence of a circle that separates two sets of points $\{(x_{r1}, y_{r1}), \ldots, (x_{rn}, y_{rn})\}$ (red) and $\{(x_{g1}, y_{g1}), \ldots, (x_{gm}, y_{gm})\}$ (green). Treat these together as a 'signed' dataset of the form $\{(x_1, y_1, s_1), \ldots, (x_{n+m}, y_{n_m}, s_{n+m})\}$ where $s_t = +1$ for the red points and $s_t = -1$ for the green ones. Transform this dataset into the 'quadratic'-space — $\{(\boldsymbol{p}_1, s_1), \ldots, (\boldsymbol{p}_{n+m}, s_{n+m})\}$ where $\boldsymbol{p}_t = (x_t^2,\ y_t^2,\ x_t.y_t,\ x_t,\ y_t,\ 1)$. A circular separator for the original dataset will be of the form $(x-c_x)^2 + (y-c_y)^2 = c \equiv x^2 + y^2 - 2c_x.x - 2c_y.y + (c_x^2 + c_y^2 - c) = 0$. This is equivalent to finding a separator of the form $(1, 1, 0, w_1, w_2, w_3) \circ \boldsymbol{v} = 0$ where $\circ$ denotes the inner product between the two vectors and $v \in \mathscr{R}^6$ with $v_6 = 1$. For this we require that for $1 \leq t \leq (n+m)$

$$s_t.\left((1, 1, 0, w_1, w_2, w_3) \circ \boldsymbol{p}_t\right) > 0 \ \equiv \ s_t.\left(x_t^2 + y_t^2 + w_1 * x_t + w_2 * y_t + w_3\right) = \epsilon_t,\ \epsilon_t > 0 \tag{1}$$

We also need to remember the fact that we need to be able to recover the separating circle after this. Assuming we get a solution based on Equations 1, mapping it back to the circle we get

$$c_x = -\frac{w_1}{2}, \quad c_y = -\frac{w_2}{2}, \quad \text{and} \quad c = c_x^2 + c_y^2 - w_3 = \frac{w_1^2 + w_2^2}{4} - w_3$$

c being the square of the radius of the separating circle, clearly we have the additional constraint that

$$w_1^2 + w_2^2 - 4.w_3 > 0 \tag{2}$$

Now notice that any matrix of the form $\begin{bmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ u_3 & v_3 & 1 \end{bmatrix}$ must be non-singular as long as the three points $(u_1, v_1), (u_2, v_2)$ and (u_3, v_3) are not collinear. Find any three points (x_i, y_i), (x_j, y_j), (x_k, y_k) that are not collinear. If the entire dataset (red and green points together) is collinear, then it is easy to check if the two sets are separable by a circle — they will be iff at least one of groups appears contiguously on the line. Having picked the three non collinear points we get

$$\begin{bmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{bmatrix} . \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} \epsilon_i - x_i^2 - y_i^2 \\ \epsilon_j - x_j^2 - y_j^2 \\ \epsilon_k - x_k^2 - y_k^2 \end{bmatrix} \Rightarrow \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{bmatrix}^{-1} . \begin{bmatrix} \epsilon_i - x_i^2 - y_i^2 \\ \epsilon_j - x_j^2 - y_j^2 \\ \epsilon_k - x_k^2 - y_k^2 \end{bmatrix}$$

This means we can write each $w_l, 1 \leq l \leq 3$ as $w_l = a_{li}.\epsilon_i + a_{lj}.\epsilon_j + a_{lk}.\epsilon_k + b_l$. Equation 1 after substitution for the w_l in terms of $\epsilon_i, \epsilon_j, \epsilon_k$ would become:

$$\sum_{l \in \{i,j,k\}} s_t.(x_t.a_{1l} + y_t.a_{2l} + a_{3l})\epsilon_l = \epsilon_t - s_t.\left(x_t^2 + y_t^2 + b_1.x_t + b_2.y_t + b_3\right), \ \ \forall 1 \leq t \notin \{i,j,k\} \leq (n+m)$$

These are $(m+n-3)$ equations in $(m+n)$ variables $\epsilon_1, \ldots, \epsilon_{n+m}$. Note that plugging in the expressions for w_l (in terms of ϵ_l) will result in identities for the equations for $t = i, j, k$. We can rewrite the above equations more legibly as

$$\alpha_{ti}.\epsilon_i + \alpha_{tj}.\epsilon_j + \alpha_{tk}.\epsilon_k + \beta_t = \epsilon_t > 0, \ \ \forall 1 \leq t \notin \{i,j,k\} \leq (n+m)$$

for constants $\alpha_{ti}, \alpha_{ti}, \alpha_{ti}, \beta_t$. So we need to check if there exist values $\epsilon_i, \epsilon_j, \epsilon_k > 0$ such that

$$\alpha_{ti}.\epsilon_i + \alpha_{tj}.\epsilon_j + \alpha_{tk}.\epsilon_k + \beta_t > 0, \ \ \forall 1 \leq t \notin \{i,j,k\} \leq (n+m) \tag{3}$$

This is easy to do with a variant of Gaussian elimination as we show below. However we also need to ensure Inequality 2 holds. This takes the form:

$$\sum_{l=1}^{2} \left(a_{li}.\epsilon_i + a_{lj}.\epsilon_j + a_{lk}.\epsilon_k + b_l\right)^2 - 4.(a_{3i}.\epsilon_i + a_{3j}.\epsilon_j + a_{3k}.\epsilon_k + b_3) > 0$$

$$\equiv h_{i1}.\epsilon_i^2 + h_{i2}.\epsilon_i + h_{j1}.\epsilon_j^2 + h_{j2}.\epsilon_j + h_{k1}.\epsilon_k^2 + h_{k2}.\epsilon_k + p_{ij}\epsilon_i\epsilon_j + p_{jk}\epsilon_j\epsilon_k + p_{ik}\epsilon_i\epsilon_k + h_0 > 0 \quad (4)$$

The last expression is a simple quadratic in $\epsilon_i, \epsilon_j, \epsilon_k$. Note that $h_{i1} = a_{1i}^2 + a_{2i}^2 > 0$. Therefore the quadratic is convex (second differential is positive). A simple lemma below will come in handy later.

Lemma 1 *For any convex function $f(\boldsymbol{x})$ defined over a convex domain $\mathscr{D}$, and a convex region $\mathscr{S} \subset \mathscr{D}$, if the $\boldsymbol{X}^*$ is the set of vertices of the convex hull ('extreme' points) of $\mathscr{S}$, then*

$$\arg\max_{\boldsymbol{x} \in \mathscr{S}} f(\boldsymbol{x}) \in \boldsymbol{X}^*$$

In other words, the maximum value of the function among the points in $\mathscr{S}$ must occur at one of the extreme points of $\mathscr{S}$.

Proof: This follows almost directly from the definition of convexity. Any point $\boldsymbol{p} \in \mathscr{S}$ can be written as $\boldsymbol{\alpha}^T.\boldsymbol{X}^*$ — convex combination of the points in $\boldsymbol{X}^*$. The following is a direct consequence of the convexity of $f(.)$.

$$\max_{\boldsymbol{x} \in \boldsymbol{X}^*} f(\boldsymbol{x}) \geq \sum_{\boldsymbol{x} \in \boldsymbol{X}^*} \alpha_{\boldsymbol{x}}.f(\boldsymbol{x}) > f\left(\boldsymbol{\alpha}^T.\boldsymbol{X}^*\right) = f(\boldsymbol{p})$$

■

The main algorithm now follows.
Step 0: If $\alpha_{ti} < 0$ for all t, l, then declare NO if any $\beta_t < 0$ else declare YES provided the ϵs can be made to satisfy inequality 4. The first part is easy to see — positive values for $\epsilon_i, \epsilon_j, \epsilon_k$ will need $\beta_t > 0$ for the inequality to hold. If all $\beta_t > 0$ then the following assignment works

$$0 < \epsilon_i = \epsilon_j = \epsilon_k < \min\left(\frac{\beta_t}{|\alpha_{ti} + \alpha_{tj} + \alpha_{tk}|}\right) = \delta$$

to ensure that the inequalities in 3 are satisfied. However for the constraints in 4, notice that the box bounded by $0 \leq \epsilon_i, \epsilon_j, \epsilon_k \leq \delta$ is a convex region and we can resort to Lemma 1. So if any combination of values for the ϵs in this box satisfies inequality 4, then one of the extreme points must. We can generate all the extreme points (8 of them) with the ϵs taking 0 or δ and check if they satisfy inequality 4. If any one of them satisfies the inequality we declare YES else we declare NO.

Step 1: Check if for some $l \in \{i, j, k\}$, $\alpha_{tl} > 0$ for all t. If some such l exists (WLG say $l = i$), declare YES else go to Step 2. Reason: to assign positive values to $\epsilon_i, \epsilon_j, \epsilon_k$ such that the Inequalities 3 are true, assign some arbitrary positive values (say 1) to ϵ_j and ϵ_k. Assign

$$\epsilon_i = \max\left(1, \max_t\left(\frac{-\left(\alpha_{tj}.\epsilon_j + \alpha_{tk}.\epsilon_k + \beta_t\right)}{\alpha_{ti}}\right) + \lambda\right) \quad (5)$$

where λ is large enough to ensure that inequality 4 is satisfied.

Step 2: Let I_N denote the set of indices $\{t : \alpha_{ti} < 0\}$. Also WLG assume $i \neq 1$. Form the set of inequalities:

$$\left(\alpha_{tj} + \frac{|\alpha_{ti}|.\alpha_{1j}}{\alpha_{1i}}\right).\epsilon_j + \left(\alpha_{tk} + \frac{|\alpha_{ti}|.\alpha_{1k}}{\alpha_{1i}}\right).\epsilon_k + \left(\beta_t + \frac{|\alpha_{ti}|.\beta_1}{\alpha_{1i}}\right) > 0, \quad \forall 1 \leq t \in I_N \tag{6}$$

Notice that these are just the inequalities obtained by adding a $\frac{|\alpha_{ti}|}{\alpha_{1i}}$ multiple of the equation for $t = 1$ to every other $t \in I_N$. This eliminates ϵ_i and also keeps the sign intact. Repeat Steps 0, 1 and 2 above to Inequalities 6. If Inequalities 6 admit a positive solution for ϵ_j, ϵ_k we can always compute a matching positive value for ϵ_i using Equation 5. Similar to Step 1, if not all the coefficients of ϵ_j across the inequalities are positive then we can get a smaller set of inequalities with only ϵ_k of the form

$$\eta_t.\epsilon_k + \theta_t > 0, \quad \text{for } t \in J_N \subset I_N \tag{7}$$

Step 3: This is just for completeness. If for some $t \in J_N$ both $\eta_t, \theta_t < 0$, then declare NO. Otherwise let $N = \{t \in J_N | \eta_t < 0\}$ and the $P = \{t \in J_N | \eta_t > 0\}$ be the sets of indices for which the corresponding η_t is negative and postive respectively. For the N set, we would require that $\epsilon_k < \min_t \left(\frac{-\theta_t}{|\eta_t|}\right)$ and for the P set it must be that $\epsilon_k > \max_t \left(\frac{-\theta_t}{\eta_t}\right)$. Therefore we declare YES if

$$\max_t \left(\frac{-\theta_t}{\eta_t}\right) < \min_t \left(\frac{-\theta_t}{|\eta_t|}\right) \Rightarrow \max_t \left(\frac{-\theta_t}{\eta_t}\right) < \epsilon_k < \min_t \left(\frac{-\theta_t}{|\eta_t|}\right)$$

and declare NO otherwise.

Complexity: The algorithm works in linear time — $O(n+m)$. Transformation in Equation 1 takes linear time (one scan through the datapoints). Finding three non-collinear points with which to eliminate w_1, w_2, w_3 and form inequalities 3 takes linear time again (again one scan with constant time to invert a (3×3) matrix). Finally each of the Steps 0,1,2,3 take one scan through the dataset each. Incidentally this algorithm can even be extended to higher dimensions with the running time blowing up again linearly with d.