Solutions to Assignment 2

September 23, 2020

1 Problem 1

When the data set is linearly independent (thus $n \leq d$), then $D = (\mathbf{y}\mathbf{y}^\top) \odot (XX^\top)$ is positive definite. First of all we set $Y = \mathtt{diag}(\mathbf{y})$ and observe that

$$D = YXX^\top Y = YXX^\top Y^\top$$

Indeed, if we set $X^\top = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}$ we get

$$\begin{aligned}
YXX^\top Y &= \begin{bmatrix} y_1 & & \\ & \ddots & \\ & & y_n \end{bmatrix} \begin{bmatrix} x_1^\top \\ \vdots \\ x_n^\top \end{bmatrix} \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 & & \\ & \ddots & \\ & & y_n \end{bmatrix} \\
&= \begin{bmatrix} y_1 y_1 x_1^\top x_1 & y_1 y_2 x_1^\top x_2 & \cdots & y_1 y_n x_1^\top x_n \\ y_2 y_1 x_2^\top x_1 & y_2 y_2 x_2^\top x_2 & \cdots & y_1 y_n x_2^\top x_n \\ \vdots & \vdots & \ddots & \vdots \\ y_n y_1 x_n^\top x_1 & y_n y_2 x_n^\top x_2 & \cdots & y_n y_n x_n^\top x_n \end{bmatrix} \\
&= \begin{bmatrix} y_1 y_1 & \cdots & y_1 y_n \\ \vdots & \ddots & \vdots \\ y_1 y_n & \cdots & y_n y_n \end{bmatrix} \odot \begin{bmatrix} x_1^\top x_1 & \cdots & x_1^\top x_n \\ \vdots & \ddots & \vdots \\ x_n^\top x_1 & \cdots & x_n^\top x_n \end{bmatrix} \\
&= yy^\top \odot (XX^\top)
\end{aligned}$$

Here $y = \begin{bmatrix} y_1 \cdots y_n \end{bmatrix}^\top$.

Since the rows of X are linearly independent and $n \leq d$ we get that X has rank n so $XX^\top \in M_{n\times n}(\mathbb{R})$ has rank n and is symmetric non-negative definite so it has no zero eigenvalue and is thus positive definite. Furthermore since $y_i = \pm 1$, Y is invertible so Sylvester's law of inertia holds and thus D has n positive eigenvalues and is positive definite.

On the other hand, when data has some linear dependence, then the matrix $XX^\top$ admits a zero eigenvalue; so does D as a consequence.

2 Problem 2

Given the inequality constraints $y_i(w^\top x_i - b)) - (1 - \xi_i) \geq 0$ and $\xi_i \geq 0$, we set up the Lagrangian

$$L(w, b, \xi, \lambda) = (1/2)w^\top w + C\sum_{i=1}^n \xi_i - \sum_{i=1}^n \lambda_i(y_i(w^\top x_i - b) - (1 - \xi_i)) - \sum_{i=1}^n \lambda_{i+n}\xi_i$$

. Before we go to the first order conditions, we simplify the algebra a bit to be able to clearly use some results from the case without soft margins (referred to as "hard margins"). In particular, we isolate the terms containing ξ_i:

$$
\begin{aligned}
L(w,b,\xi,\lambda) &= (1/2)w^\top w + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\lambda_i(y_i(w^\top x_i - b) - (1-\xi_i)) - \sum_{i=1}^{n}\lambda_{i+n}\xi_i \\
&= (1/2)w^\top w - \sum_{i=1}^{n}\lambda_i(y_i(w^\top x_i - b) - 1 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\lambda_i\xi_i - \sum_{i=1}^{n}\lambda_{i+n}\xi_i \\
&= (1/2)w^\top w - \sum_{i=1}^{n}\lambda_i(y_i(w^\top x_i - b) - 1) + \sum_{i=1}^{n}(C - \lambda_i - \lambda_{i+n})\xi_i \\
&= f(w,b,\lambda) + g(\xi,\lambda)
\end{aligned}
$$

Here f is the Lagrangian from the case with hard margins. Taking derivatives with respect to w, b and ξ we get that

$$
\begin{aligned}
\frac{\partial L}{\partial w} &= w - \sum_{i=1}^{n}\lambda_i y_i x_i = 0 \\
\frac{\partial L}{\partial b} &= \sum_{i=1}^{n}\lambda_i y_i = 0 \\
\frac{\partial L}{\partial \xi_k} &= C - \lambda_k - \lambda_{k+n} = 0
\end{aligned}
$$

The first two constraints are idenical to the hard margins case, keeping in mind that we only use the first n Lagrange multipliers. Then plugging in $w^* = \sum_{i=1}^{n}\lambda_i y_i x_i$ we have that

$$L(w^*,b^*,\xi^*,\lambda) = f(w^*,b^*,\lambda) + g(\xi^*,\lambda)$$

$$= -(1/2)\lambda^\top D'\lambda + [1_{1\times n}0_{1\times n}]\lambda + \sum_{i=1}^{n}(C - \lambda_i - \lambda_{i+n})\xi_i = -(1/2)\lambda^\top D'\lambda + [1_{1\times n}0_{1\times n}]\lambda$$

.

Here

$$D' = \begin{bmatrix} (yy^\top)\odot(XX^\top) & \\ & 0_{n\times n} \end{bmatrix}.$$

Furthermore the last equality follows as the g term vanishes due to the first order condition for $C - \lambda_k - \lambda_{k+n} = 0$. Thus, the dual problem for the soft margins case is

$$\max (1/2)\lambda^\top D'\lambda + [1_{1\times n}0_{1\times n}]\lambda$$

$$\text{w.r.t}\ \lambda_i \geq 0\ \forall\ 1 \leq i \leq 2n$$

$$\lambda_k + \lambda_{k+n} = C\ \forall\ 1 \leq k \leq n$$

3 Problem 3

Suppose we have $\begin{bmatrix} \widetilde{H} & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} -p \\ \lambda \end{bmatrix} = \begin{bmatrix} -\widetilde{H}p + A^\top \lambda \\ Ap \end{bmatrix} = \begin{bmatrix} \nabla f(x) \\ 0 \end{bmatrix}$. Then $Ap = 0$. Furthermore, $p^\top(-\widetilde{H}p + A^\top\lambda) = -p^\top \widetilde{H}p + p^\top A^\top \lambda = -p^\top \widetilde{H}p + (Ap)^\top \lambda = -p^\top \widetilde{H}p = p^\top \nabla f(x)$. But we set $\widetilde{(H)}$ to be positive definite so $p^\top \widetilde{H}p > 0$. Thus $p^\top \nabla f(x) = -p^\top \widetilde{H}p < 0$.

4 Problem 4

The non-linear 2-D embedding is given by

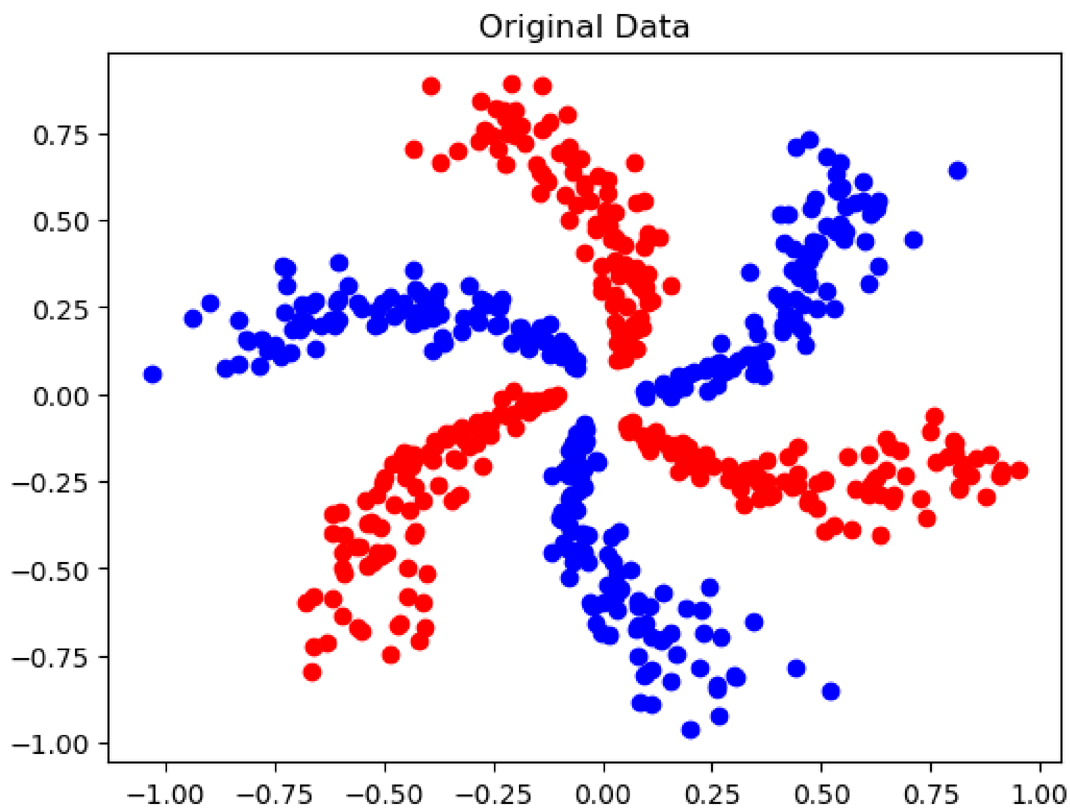
$$\varphi(x, y) = \left(\cos\left(3\left(\arctan(y/x) - \sqrt{x^2 + y^2}\right)\right), \sin\left(3\left(\arctan(y/x) - \sqrt{x^2 + y^2}\right)\right)\right)$$

In polar coordinates, $\varphi(x(r,\theta), y(r,\theta)) = (\cos(3(\theta - r)), \sin(3(\theta - r)))$.

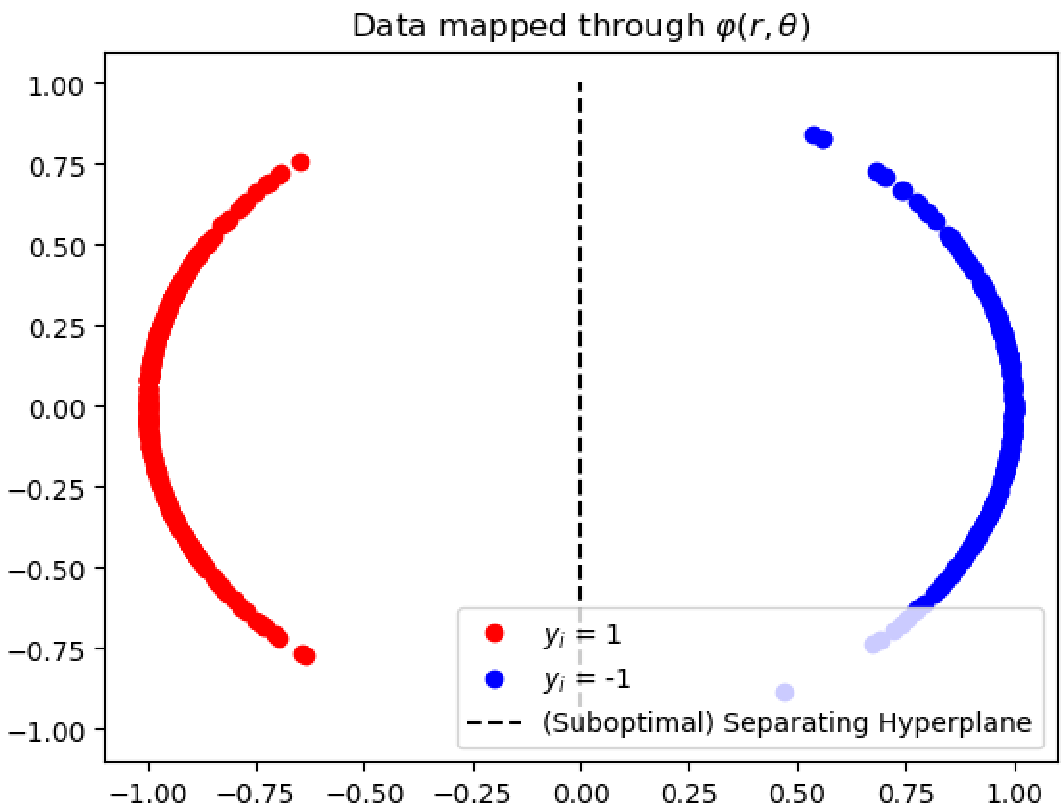
```
In [1]: using LinearAlgebra
        using PyPlot
        using DelimitedFiles
```

```
 Warning: PyPlot is using tkagg backend, which is known to cause crashes on MacOS (#410); use th
 @ PyPlot /Users/shashanksule/.julia/packages/PyPlot/4wzW1/src/init.jl:192
```

```
In [47]: Data = readdlm("stardata.txt");
         y1 = A[A[:,3] .> 0,:];
         y2 = A[A[:,3] .< 0,:]
         plot(y1[:,1],y1[:,2], "ro");
         plot(y2[:,1,], y2[:,2], "bo");
         title("Original Data");
```



In [105]:
```
 = atan.(Data[:,2],Data[:,1]);
r = sqrt.((Data[:,1]).^2 + (Data[:,2]).^2);
Embedding = [cos.(3 .*( .- r)) sin.(3 .*( .- r)) Data[:,3]];
Cluster1 = Embedding[Embedding[:,3] .> 0,:];
Cluster2 = Embedding[Embedding[:,3] .< 0,:];
Line = [zeros(length()) LinRange(-1,1,length())];
plot(Cluster1[:,1], Cluster1[:,2], "ro");
plot(Cluster2[:,1], Cluster2[:,2], "bo");
plot(Line[:,1], Line[:,2], "k--");
title(L"Data mapped through $\varphi(r,\theta)$");
legend([L"$y_i$ = 1", L"$y_i$ = -1", "(Suboptimal) Separating Hyperplane"], loc = 4, f
```


Data mapped through $\varphi(r, \theta)$
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00
y_i = 1
y_i = -1
(Suboptimal) Separating Hyperplane