Distance Weighted Discrimination

Adam Rahman

September 2, 2017

Given two sets of points in a matrix $\mathbf{X} \in \mathcal{R}^n$ with associated class variables [-1,1] in $\mathbf{Y} = diag(\mathbf{y})$, distance weighted discrimination ([1]) seeks to classify the points into two distinct subsets by finding a hyperplane between the two sets of points. Mathematically, the distance weighted discrimination problem seeks a hyperplane defined by a normal vector, $\boldsymbol{\omega}$, and position, β, such that each element in the residual vector $\bar{\mathbf{r}} = \mathbf{Y}\mathbf{X}^\mathsf{T}\boldsymbol{\omega} + \beta\mathbf{y}$ is positive and large. Since the class labels are either 1 or -1, having the residuals be positive is equivalent to having the points on the proper side of the hyperplane.

Of course, it may be impossible to have a perfect separation of points using a linear hyperplane, so an error term ξ is introduced. Thus, the perturbed residuals are defined to be

$$\mathbf{r} = \mathbf{Y}\mathbf{X}^\mathsf{T}\boldsymbol{\omega} + \beta\mathbf{y} + \boldsymbol{\xi}$$

Distance Weighted Discrimination solves the following optimization problem to find the optimal hyperplane[1].

$$\begin{array}{ll} \underset{\mathbf{r},\ \boldsymbol{\omega},\ \beta,\ \boldsymbol{\xi}}{\text{minimize}} & \sum_{i=1}^{n}(1/r_i) + C\mathbf{1}^\mathsf{T}\boldsymbol{\xi} \\ \text{subject to} & \\ & \begin{array}{rcl} \mathbf{r} & = & \mathbf{Y}\mathbf{X}^\mathsf{T}\boldsymbol{\omega} + \beta\mathbf{y} + \boldsymbol{\xi} \\ \boldsymbol{\omega}^\mathsf{T}\boldsymbol{\omega} & \leq & 1 \\ \mathbf{r} & \geq & \mathbf{0} \\ \boldsymbol{\xi} & \geq & \mathbf{0} \end{array} \end{array}$$

where $C > 0$ is a penalty parameter to be chosen.

The function `dwd` takes as input two $n \times p$ matrices `X1` and `X2` containing the points to be separated, as well as a penalty term `C` ≥ 0 penalizing the movement of a point on the wrong side of the hyperplane to the proper side, and returns the input variables necessary for `sqlp` to solve the distance weighted discrimination problem.

```
R> out <- dwd(X1,X2,C)
R> blk <- out$blk
R> At <- out$At
R> C <- out$C
R> b <- out$b

R> sqlp(blk,At,C,b)
```

Numerical Example

Consider two point configurations - `An` and `Ap` - which we would like to classify using distance weighted discrimination. Each point configuration is a matrix containing 50 points in three dimensional space.

```
R> data(Andwd)
R> data(Apdwd)
```

```
R> d <- ncol(Andwd)

R> head(Andwd)

         V1     V2     V3
[1,]  0.214 -1.577 -1.525
[2,]  0.480  0.624 -0.501
[3,]  0.088  0.330 -1.213
[4,]  0.444 -0.398 -0.630
[5,] -0.363 -1.081 -1.447
[6,]  0.123 -0.077 -0.167

R> head(Apdwd)

         V1     V2    V3
[1,] -0.687  0.192 0.726
[2,]  0.444  0.782 0.887
[3,]  2.360 -1.114 0.089
[4,]  2.230  1.428 1.369
[5,]  1.555 -0.142 2.138
[6,]  0.259  0.163 1.818
```

Distane weighed discrimination is used to seperate these two configrurations by specifying an appropriate penalization term. Here, we will take a value of 0.5.

```
R> out <- dwd(Apdwd,Andwd,0.5)
R> blk <- out$blk
R> At <- out$At
R> C <- out$C
R> b <- out$b

R> out <- sqlp(blk, At, C, b)
```

The information defining the seperating hyperplane (ω and β) is stored in the `X` output vector.

```
X <- out$X

omega <- X[[1]][2:(d+1)]
beta <- X[[1]][d+3]

omega
          [,1]
[1,] 0.6567689
[2,] 0.4857645
[3,] 0.5767907

beta

[1] -0.7520769
```

References

[1] James Stephen Marron, Michael J Todd, and Jeongyoun Ahn. Distance-weighted discrimination. *Journal of the American Statistical Association*, 102(480):1267–1271, 2007.