Beyond Hard Negative Mining: Efficient Detector Learning via Block-Circulant Decomposition

João F. Henriques, João Carreira, Rui Caseiro and Jorge Batista
Institute of Systems and Robotics, University of Coimbra
{henriques,joaoluis,ruicaseiro,batista}@isr.uc.pt

A. Proofs

A.1. Separability of learning problems

For the exact case, we will assume D can be written in terms of dot-products, as mentioned. Keeping in mind that its arguments are scalar,

$$D\left(\alpha_i, y_i\right) = d_1 \left\|\alpha_i\right\|^2 + d_2 \left\|y_i\right\|^2 + d_3 \alpha_i^H y_i, \tag{A.1}$$

where we defined the new constants d_1, d_2 and d_3. Then, Eq. 2 becomes

$$\begin{aligned} &\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^H G \boldsymbol{\alpha} + \sum_{i=1}^{n}\left(d_1 \left\|\alpha_i\right\|^2 + d_2 \left\|y_i\right\|^2 + d_3 \alpha_i^H y_i\right) \\ &= \min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^H G \boldsymbol{\alpha} + d_1 \left\|\boldsymbol{\alpha}\right\|^2 + d_2 \left\|\mathbf{y}\right\|^2 + d_3 \boldsymbol{\alpha}^H \mathbf{y}, \end{aligned} \tag{A.2}$$

where the Hermitian transpose $(\cdot)^H$ is used instead of $(\cdot)^T$. Notice that all the learning algorithms we consider use the Hermitian transpose when extended from reals to complex numbers, which simplifies some expressions, and has no effect if the quantities are indeed real.

Performing the substitutions $G = U^{-1}\bar{G}U$, $\boldsymbol{\alpha} = U^{-1}\bar{\boldsymbol{\alpha}}$ and $\mathbf{y} = U^{-1}\bar{\mathbf{y}}$, by unitarity the U's cancel out, and we are left with

$$\min_{\bar{\boldsymbol{\alpha}}} \frac{1}{2}\bar{\boldsymbol{\alpha}}^H \bar{G} \bar{\boldsymbol{\alpha}} + d_1 \left\|\bar{\boldsymbol{\alpha}}\right\|^2 + d_2 \left\|\bar{\mathbf{y}}\right\|^2 + d_3 \bar{\boldsymbol{\alpha}}^H \bar{\mathbf{y}}. \tag{A.3}$$

This may seem like a trivial change, but $\bar{G}$ is block-diagonal, while G is not. Recall Eq. 13 (which we replicate here to make $\bar{G}$'s structure more clear),

$$\bar{G} = \begin{bmatrix} \bar{G}(1) & & & \\ & \bar{G}(2) & & \\ & & \ddots & \\ & & & \bar{G}(s) \end{bmatrix}, \tag{A.4}$$

where each block is $n \times n$. To correspond to the same structure, split the vectors $\bar{\boldsymbol{\alpha}}$ and $\bar{\mathbf{y}}$ into s blocks of size $n \times 1$,

$$\bar{\boldsymbol{\alpha}} = \begin{bmatrix} \bar{\boldsymbol{\alpha}}_1 \\ \bar{\boldsymbol{\alpha}}_2 \\ \vdots \\ \bar{\boldsymbol{\alpha}}_s \end{bmatrix}, \quad \bar{\mathbf{y}} = \begin{bmatrix} \bar{\mathbf{y}}_1 \\ \bar{\mathbf{y}}_2 \\ \vdots \\ \bar{\mathbf{y}}_s \end{bmatrix}, \tag{A.5}$$

where each block $\bar{\boldsymbol{\alpha}}_f$ and $\bar{\mathbf{y}}_f$ has n elements $\bar{\boldsymbol{\alpha}}_{fi}$ and $\bar{\mathbf{y}}_{fi}$, respectively. Now, since the rules for matrix products are the same as for block-matrix products, direct computation yields

$$\begin{aligned} &\min_{\bar{\boldsymbol{\alpha}}} \sum_{f=1}^{s}\left(\frac{1}{2}\bar{\boldsymbol{\alpha}}_f^H \bar{G}_f \bar{\boldsymbol{\alpha}}_f + d_1 \left\|\bar{\boldsymbol{\alpha}}_f\right\|^2 + d_2 \left\|\bar{\mathbf{y}}_f\right\|^2 + d_3 \bar{\boldsymbol{\alpha}}_f^H \bar{\mathbf{y}}_f\right) \\ &= \min_{\bar{\boldsymbol{\alpha}}} \sum_{f=1}^{s}\left(\frac{1}{2}\bar{\boldsymbol{\alpha}}_f^H \bar{G}_f \bar{\boldsymbol{\alpha}}_f + \sum_{i=1}^{n} D\left(\bar{\alpha}_{fi}, \bar{y}_{fi}\right)\right). \end{aligned} \tag{A.6}$$

Notice that Eq. A.6 is a sum of objective functions over different (and non-interacting) optimization variables, $\bar{\boldsymbol{\alpha}}_f$. As such, they can be optimized independently, and Eq. A.6 is equivalent to the s sub-problems,

$$\min_{\bar{\boldsymbol{\alpha}}_f} \frac{1}{2}\bar{\boldsymbol{\alpha}}_f^H \bar{G}_f \bar{\boldsymbol{\alpha}}_f + \sum_{i=1}^{n} D\left(\bar{\alpha}_{fi}, \bar{y}_{fi}\right), \tag{A.7}$$

for $f = 1, \ldots, s$, as required.

A.1.1 Remark on transformation matrices that yield exact decompositions

As an interesting aside, it is possible to characterize the class of matrices U that would yield an *exact* decomposition for most algorithms. Just as unitary matrices preserve the L^2-norm, it is known [1] that generalized permutation matrices (which extend permutation matrices by allowing the non-zero elements to take the values 1 and -1) preserve all L^p-norms, for $p \geq 1$. By restricting U to this class, the decomposition would be exact for algorithms such as the SVR. Even though this result may be useful for other block-diagonalizations, we cannot use it since it is too restrictive for the case of block-circulant matrices.

A.2. Complex Support Vector Regression

This section deals with the extension of a linear regression problem,

$$\min_{\mathbf{w}} \|\mathbf{w}\|^2 + c\sum_{j=1}^{n} \left|\mathbf{w}^H\mathbf{x} - y_j\right|_\epsilon, \tag{A.8}$$

to the complex domain, through the extended loss function,

$$\left|\mathbf{w}^H\mathbf{x} - y\right|_\epsilon = \left|\mathrm{Re}\left(\mathbf{w}^H\mathbf{x} - y\right)\right|_\epsilon + \left|\mathrm{Im}\left(\mathbf{w}^H\mathbf{x} - y\right)\right|_\epsilon, \tag{A.9}$$

as mentioned in Section 5. Note the Hermitian transpose reduces to the transpose for real arguments. Also, the ϵ-insensitive loss in Eq. A.8 and A.9 can be easily squared, in the case of L2-SVR, and our result still holds.

In this section we will use i to denote a pure imaginary unit, $i = \sqrt{-1}$. First, decompose all quantities into their real and imaginary components, as

$$\begin{aligned} \mathbf{w} &= \mathbf{w}^R + i\mathbf{w}^I \\ \mathbf{x}_j &= \mathbf{x}_j^R + i\mathbf{x}_j^I \\ y_j &= y_j^R + iy_j^I. \end{aligned} \tag{A.10}$$

Substituting into Eq. A.8, and applying the rules of the complex product,

$$\begin{aligned} \min_{\mathbf{w}} \left\|\mathbf{w}^R + i\mathbf{w}^I\right\|^2 + c\sum_{j=1}^{n} \Big| \left(\mathbf{w}^R\right)^T \mathbf{x}_j^R + \left(\mathbf{w}^I\right)^T \mathbf{x}_j^I - y_j^R \\ + i\left(\left(\mathbf{w}^R\right)^T \mathbf{x}_j^I - \left(\mathbf{w}^I\right)^T \mathbf{x}_j^R - y_j^I\right)\Big|_\epsilon. \end{aligned} \tag{A.11}$$

Expanding the first term, and applying Eq. A.9 to the second,

$$\begin{aligned} \min_{\mathbf{w}} \left\|\mathbf{w}^R\right\|^2 + c\sum_{j=1}^{n} \left| \left(\mathbf{w}^R\right)^T \mathbf{x}_j^R + \left(\mathbf{w}^I\right)^T \mathbf{x}_j^I - y_j^R \right|_\epsilon \\ + \left\|\mathbf{w}^I\right\|^2 + c\sum_{j=1}^{n} \left| \left(\mathbf{w}^R\right)^T \mathbf{x}_j^I - \left(\mathbf{w}^I\right)^T \mathbf{x}_j^R - y_j^I \right|_\epsilon. \end{aligned} \tag{A.12}$$

Eq. A.12 shows that the complex SVR is equivalent to an augmented real SVR, with:

1. Double the features, to account for real and imaginary parts of the inputs and weights.
2. Double the samples, to account for the loss function in the real axis and in the imaginary axis (from Eq. A.9).

The augmented real SVR can be written more compactly as

$$\min_{\mathbf{w}} \left\|\mathbf{w}'\right\|^2 + c\sum_{j=1}^{n} \left|\mathbf{w}'^T\mathbf{x}_j' - y_j'\right|_\epsilon, \tag{A.13}$$

with $\mathbf{x}_j'$ the rows of X' and y_j' the elements of $\mathbf{y}'$, defined in terms of the analogous original complex quantities,

$$\begin{aligned} X' &= \begin{bmatrix} \mathrm{Re}\left(X\right) & \mathrm{Im}\left(X\right) \\ \mathrm{Im}\left(X\right) & -\mathrm{Re}\left(X\right) \end{bmatrix} \\ \mathbf{y}' &= \begin{bmatrix} \mathrm{Re}\left(\mathbf{y}\right) \\ \mathrm{Im}\left(\mathbf{y}\right) \end{bmatrix} \\ \mathbf{w}' &= \begin{bmatrix} \mathrm{Re}\left(\mathbf{w}\right) \\ \mathrm{Im}\left(\mathbf{w}\right) \end{bmatrix}. \end{aligned} \tag{A.14}$$

References

[1] C.-K. Li and W. So. Isometries of lp-norm. *The American Mathematical Monthly*, 101(5):452, May 1994. A.1.1