The Spatial LASSO with Applications to Unmixing Hyperspectral Biomedical Images - Supplemental Material

Daniel V. Samarov, Maritoni Litorja, Jeeseong Hwang *

October 1, 2014

S1 Supplemental Material

This document provides additional details on the Spatial LASSO (SPLASSO) model discussed in Samarov et al. (2013). Section S2 details several properties of the SPLASSO, Section S3 provides a blockwise-coordinate descent approach to solving the SPLASSO, Section S4 introduces the adaptive splasso, Section **??** highlights commonalities with microarray processing and Section S5 provides proofs.

*Daniel V. Samarov is a Mathematical Statistician, Statistical Engineering Division, Information Technology Laboratory, National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899, (Email: daniel.samarov@nist.gov), Maritoni Litorja is a Research Chemist, Sensor Science Division, Physical Measurement Laboratory (PML), NIST, Gaithersburg, MD 20899, (Email: litorja@nist.gov), Jeeseong Hwang is a Research Biophysicist, Radiation and Biomolecular Physics Division, PML, NIST, Gaithersburg, MD 20899, (Email: jch@nist.gov).

S2 SPLASSO properties

S2.1 Derivation of the orthonormal case

In Section 3.2 of Samarov et al. (2013) the SPLASSO is studied under the assumption of orthonormality of the design matrix. In solving for $\boldsymbol{\beta}_i$ in the orthonormal setting we start by fixing the remaining $j \neq i$ coefficients. This is the same first step taken in the block-wise coordinate descent based procedure we use to solve the SPLASSO problem (discussed in Section 4) where all but the current parameter being estimated fixed.

Starting with equation (3.4) this can be expanded to

$$\sum_{l'=1}^{d}\left(\beta_{il'}^2 - \mathbf{y}_i^T\mathbf{x}_l'\beta_{il'} + \lambda_1|\beta_{il'}| + \lambda_2\sum_{j\in N_k(\mathbf{y}_i)}(\beta_{il'}^2 - 2\beta_{il'}\beta_{jl'})\right) + \mathbf{y}_i^T\mathbf{y}_i + \lambda_2\sum_{j\in N_k(\mathbf{y}_i)}\boldsymbol{\beta}_j^T\boldsymbol{\beta}_j. \tag{S2.1}$$

Noting that the objective function in (3.4) is convex, the optimal solution in (S2.1) is characterized by its subgradient equations (as the l_1 penalty term is non-differentiable). Recall from our discussion in Section 3.2 that for illustrative purposes we assume $\sum_{j\in N_k(\mathbf{y}_i)} w_{ij} = 1$ and that $\alpha_{il} = \sum_{j\in N_k(\mathbf{y}_i)}\beta_{jl}w_{ij}$ and $\hat{\beta}_{il}(\text{OLS}) = \mathbf{y}_i^T\mathbf{x}_l$. With these notations and assumptions the subgradient equations for a particular $\hat{\beta}_{il}$ are written as

$$\hat{\beta}_{il}(\text{OLS}) - (1+\lambda_2)\hat{\beta}_{il} + \lambda_2\alpha_{il} = \frac{\lambda_1}{2}v_l \tag{S2.2}$$

where the subderivative term

$$v_l = \begin{cases} \text{sgn}(\hat{\beta}_{il}) & \text{if } \hat{\beta}_{il} \neq 0, \\ \in \{v_l : |v_l| \leq 1\} & \text{if } \hat{\beta}_{il} = 0. \end{cases}$$

Dividing (S2.2) by $1+\lambda_2$, letting $\gamma = 1/(1+\lambda_2)$ and setting $\hat{b}_{il} = \gamma\hat{\beta}_{il}(\text{OLS}) + (1-\gamma)\alpha_{il}$ the

expression in (S2.2) can be written as

$$\hat{b}_{il} - \hat{\beta}_{il} = \frac{\lambda_1}{2}\gamma. \tag{S2.3}$$

When $\hat{\beta}_{il} = 0$, the subgradient equations are satisfied when

$$|\hat{b}_{il}| \leq \frac{\lambda_1}{2}\gamma$$

and when $\hat{\beta}_{il} \neq 0$ we have

$$\hat{\beta}_{il} = \text{sgn}(\hat{b}_{il})(|\hat{b}_{il}| - \frac{\lambda_1}{2}\gamma).$$

Note, due to the convexity and continuity of the objective function in (3.4) when $\hat{\beta}_{il} \neq 0$, $\text{sgn}(\hat{b}_{il}) = \text{sgn}(\hat{\beta}_{il})$. Putting this all together, for fixed $j \neq i$ the SPLASSO estimate in the orthonormal setting is

$$\hat{\beta}_{il}(\text{SPLASSO}) = \text{sgn}(\hat{b}_{il})(|\hat{b}_{il}| - \frac{\lambda_1}{2}\gamma)_+. \tag{S2.4}$$

S2.2 The Orthonormal Case - Additional Discussion

Following the discussion from Section 3.2 in Samarov et al. (2013), Figure 1 compares the coefficient estimates of the OLS, LASSO and SPLASSO estimators under the assumption of orthonormality and varying parameter values. Here we set the weighted average of the surrounding coefficients, $\alpha_{i,l}$ to be $\{0.1, 1, 2\}$ (increasing from left to right along the columns) with the regularization parameters $\lambda_1 = 1$ and λ_2 taking values in $\{0.1, 1, 2\}$ (increasing from top to bottom along the rows). The 45° line corresponds to the OLS estimate, the bold solid line is the LASSO estimate, the dashed bold line the SPLASSO estimate and the solid black point is the value of $\alpha_{i,l}$.

We start off by looking at the effect that the $\alpha_{i,l}$'s have on the SPLASSO estimate. Going

from left to right along the columns of plots in Figure 1 one can see that as the weighted average of the surrounding coefficients increases, the region over which thresholding occurs begins to shift over to the left (this is particularly prominent in the second and third row of plots). The implication is that unless the current coefficient's estimated value is very different from its neighbors, it is less likely to be set to 0 and more likely to be positive.

Next, as the value of the regularization parameter λ_2 increases (going from top to bottom) it has the effect of placing greater weight on the $\alpha_{i,l}$s. This also results in a shift left in the region where the SPLASSO estimates are set to be 0.

S2.3 Feasible Set

Looking at the set of feasible solutions for the SPLASSO in the general case is important as it gives us an understanding of how the regularization parameters, value of neighboring coefficients and the spatial weights effect the coefficient estimates. Because the value of the spatial penalty term depends on the neighboring coefficients, we consider a few test cases to get a general idea of the models behavior. Consider the following example; looking at Figure 2 suppose we are estimating the coefficient vector $\boldsymbol{\beta} \in \mathbb{R}^2$ corresponding to the white square at the center. The $k = 1$ neighborhood of this point is indicated by the light gray and dark gray squares whose respective coefficient values are identical and are denoted by $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$. Lastly let w_1 and w_2 be the spatial weights for $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$, respectively. Table 1 details the test cases we consider here. The particular parameterizations chosen are meant to reflect what might be expected in practice. For the regularization parameters we consider the cases where $\lambda_1 = \lambda_2$, $\lambda_1 < \lambda_2$ or $\lambda_1 > \lambda_2$ with either all the surrounding coefficients close in value ($\boldsymbol{\beta}_1 \approx \boldsymbol{\beta}_2$), or with one set greater than the other ($\boldsymbol{\beta}_1 > \boldsymbol{\beta}_2$), and having either similar spatial weights ($w_1 \approx w_2$), or with the weights greater for one set of coefficients than the other ($w_1 > w_2$ or $w_1 < w_2$). The feasible sets corresponding to each of these cases are shown in Figure 3.

For example, in the Case I in Table 1 we have $\boldsymbol{\beta}_1 = \boldsymbol{\beta}_2 = (1/2, 1/2)$, $w_1 = w_2 = 1/2$ and $\lambda_1 = \lambda_2 = 1$. This case captures the SPLASSO estimates behavior when all the surrounding

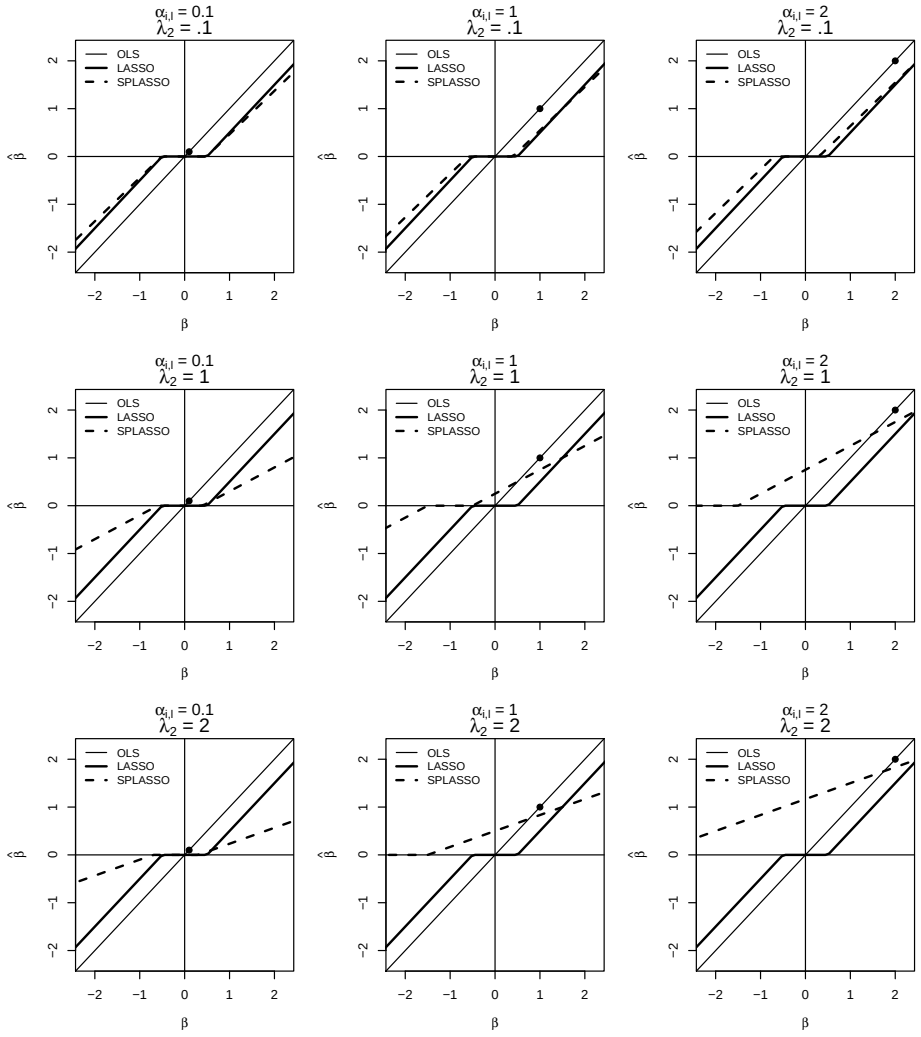

Figure 1: An illustration of the behavior of the OLS, LASSO and SPLASSO coefficient estimates in the orthonormal setting. Looking across the plots the values of the surrounding neighbors, $\alpha_{i,l}$ and regularization parameter λ_2 steadily increase as we move left to right and top to bottom.

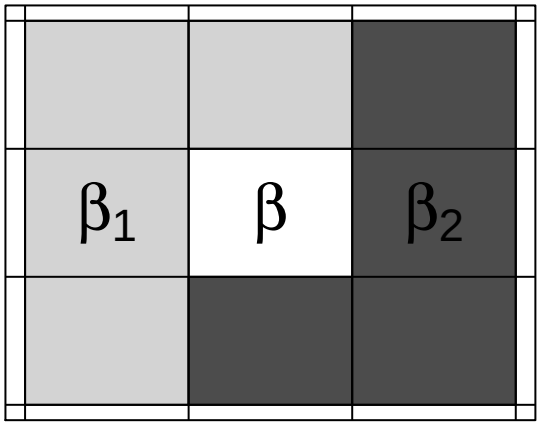

Figure 2: An illustration of the test cases we consider for exploring the feasible set of the SPLASSO model. The white "pixel" at the center corresponds to the location whose coefficients (or abundances in the context of HSI), $\boldsymbol{\beta}$ we are estimating and the dark and light gray pixels are its neighbors. The light and dark gray regions each share common coefficients vectors $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ respectively.

coefficients and weights on the sparse and spatial penalties are equal. In cases V and IX once again $\boldsymbol{\beta}_1 = \boldsymbol{\beta}_2$ but now we consider the cases where either the sparse term is less than the spatial term, $\lambda_1(=1) < \lambda_2(=2)$ or visa-versa, $\lambda_1(=2) > \lambda_2(=1)$. The other cases shown in Table 1 consider other possible combinations of neighboring coefficients, regularization and spatial weight values. Note, we do not claim that the cases considered here are exhaustive, as stated earlier they are meant to provide insight into general model behavior. Figure 3 shows the feasible sets, highlighted in light blue, and Table 1 provides details on the parameters used.

Looking at Figure 3 we can see that depending on whether λ_1 is equal, greater or less than λ_2 (going from left to right) the corners of the feasible set become more or less pronounced and likely to have thresholding occur. Similarly, depending on whether $\boldsymbol{\beta}_1$ is equal to or greater than $\boldsymbol{\beta}_2$ and whether the spatial weight w_1 is equal, greater than or less than w_2, we see that the feasible set is pulled more or less in the direction of the parameter with the greatest weight.

Case	I	II	III	IV	V	VI	VII	VIII	IX
$\boldsymbol{\beta}_1$	$(\frac{1}{2},\frac{1}{2})$	(1,1)	(1,1)	(1,1)	$(\frac{1}{2},\frac{1}{2})$	(1,1)	(1,1)	(1,1)	$(\frac{1}{2},\frac{1}{2})$
$\boldsymbol{\beta}_2$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$
w_1	$\frac{1}{2}$	$\frac{1}{2}$	0.75	0.25	$\frac{1}{2}$	$\frac{1}{2}$	0.75	0.25	$\frac{1}{2}$
w_2	$\frac{1}{2}$	$\frac{1}{2}$	0.25	0.75	$\frac{1}{2}$	$\frac{1}{2}$	0.25	0.75	$\frac{1}{2}$
λ_1	1	1	1	1	1	1	1	1	2
λ_2	1	1	1	1	2	2	2	2	1

Case	X	XI	XII
$\boldsymbol{\beta}_1$	(1,1)	(1,1)	(1,1)
$\boldsymbol{\beta}_2$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$	$(\frac{1}{2},\frac{1}{2})$
w_1	$\frac{1}{2}$	0.75	0.25
w_2	$\frac{1}{2}$	0.25	0.75
λ_1	2	2	2
λ_2	1	1	1

Table 1: The values used to generate the different feasible regions shown in Figure 3. The columns, labelled I, II, ..., XII correspond the coefficients used in each of the panels shown in Figure 3. The values chosen were meant to reflect the cases one might expect to see in practice.

S2.4 Decorrelation, convexity and relationship to the elastic net

As previously mentioned, the SPLASSO penalty shares some similarities with the elastic net, in particular it combines the "decorrelation" property of the latter with spatial smoothing across the coefficients. Before exploring this connection in more detail we introduce some notation; let $d_i = \sum_{j=1}^{n} w_{ij}$, with $w_{ij} = w_{ji}$, $\mathbf{D} = \text{diag}(d_1, \ldots, d_n)$ and $\mathcal{D} = \mathbf{D} \otimes \mathbf{I}_{m\times m}$, where $\otimes$ denotes the Kronecker product. Next define $\mathbf{W} = \{w_{ij}\}_{i,j=1}^{n}$, $\mathcal{W} = \mathbf{W} \otimes \mathbf{I}_{m\times m}$ and $\mathcal{L} = \mathcal{D} - \mathcal{W}$. Finally, letting $\mathcal{B} = (\boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_n^T)^T$, $\mathcal{X} = \mathbf{I}_{n\times n} \otimes \mathbf{X}$ and $\mathcal{Y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_n^T)^T$ we can express the SPLASSO objective function as

$$\hat{\mathcal{B}}(\text{naïve SPLASSO}) = \arg\min_{\mathcal{B}} ||\mathcal{Y} - \mathcal{X}\mathcal{B}||^2 + \lambda_1|\mathcal{B}|_1 + \lambda_2\mathcal{B}^T\mathcal{L}\mathcal{B}. \tag{S2.5}$$

This representation of the SPLASSO loss function allows for some interesting insights. The key observation is that the matrix $\mathcal{L}$ in (S2.5) is the well known "Graph Laplacian" used in spectral clustering (see von Luxburg (2007) for an overview on this topic).

Spectral clustering is an unsupervised learning method which is used as an approximation to the graph cut problem (see Hagen & Kahng (1992) and Shi & Malik (2000)). Its objective

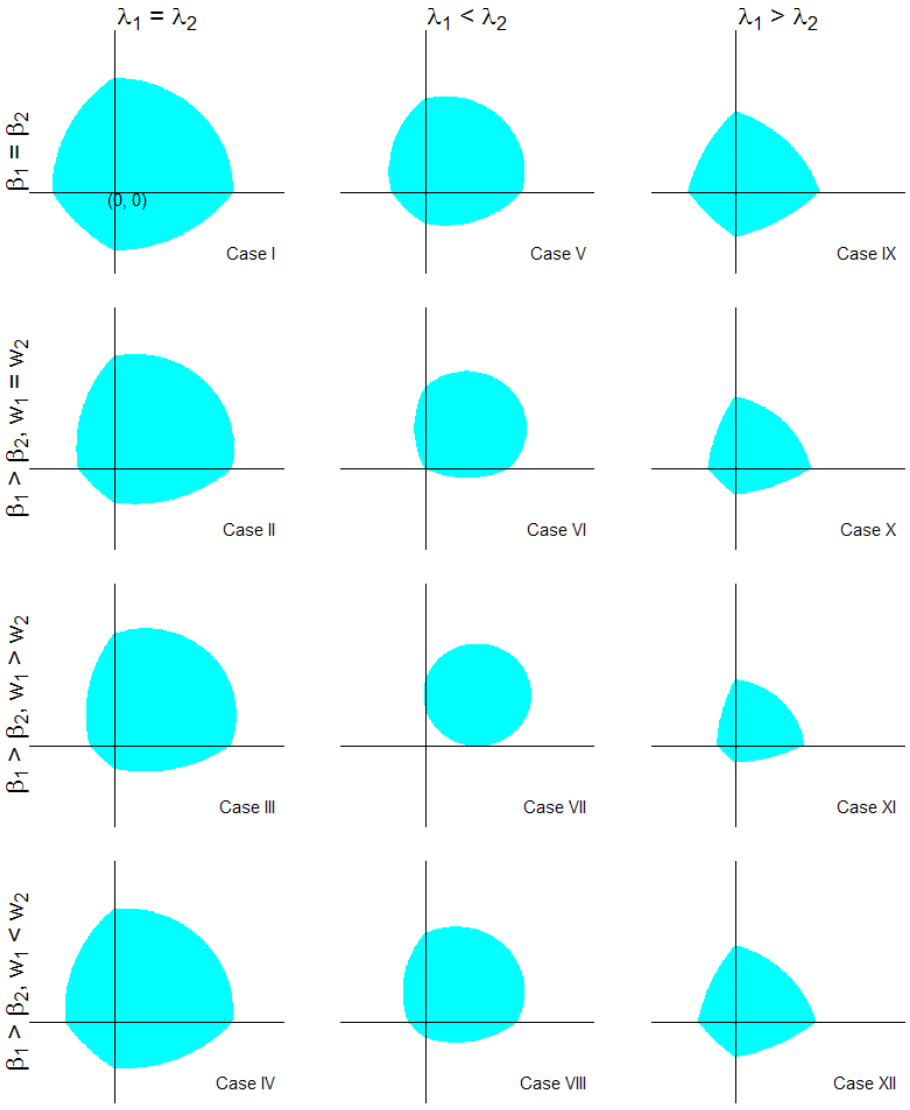

Figure 3: This figure shows the set of feasible solutions for the SPLASSO model and its relationship to the regularizations parameters λ_1 and λ_2 (columns) and the neighboring coefficients and their associated weights (rows). Specific values for these parameters are shown in Table 1.

is to separate (i.e. cluster) a collection of observations into two or more groups based on the assumption that there exist high and low density regions in the data. More specifically, spectral clustering looks at each observations as a node on a graph with the graph Laplacian matrix representing the edge weights. If the edge weights are properly specified the points which are similar to one another will have larger edge weights (i.e. will be strongly connected to one another), and those that are different will have smaller edge weights (i.e.

are less connected). The final step is to compute the eigenvalues and vectors of the graph Laplacian and apply a standard clustering algorithm (such as k-means) to the eigenvectors corresponding to the $k-1$ smallest eigenvalues (excluding the smallest) to determine the k clusters. For a discussion on the intuition behind the latter we refer the reader to von Luxburg (2007).

Expressing the spatial penalty in the SPLASSO objective function using the graph Laplacian provides us with some interesting insights. First, the positive semi-definiteness of the graph Laplacian ensures convexity of the SPLASSO loss. Second, in the same way that $\mathcal{L}$ connects and separates high and low density regions in the clustering setting, within the framework of our model it connects and separates (dis)similar coefficient vectors, producing smoother, less noisy estimates.

The following theorem gives a more detailed look at how our coefficient estimates are affected by the graph Laplacian.

Theorem S2.1. *Given the data* $(\mathcal{Y}, \mathcal{X})$ *and regularization parameters* (λ_1, λ_2)*, the SPLASSO estimates* $\hat{\mathcal{B}}$ *are given by*

$$\hat{\mathcal{B}}(SPLASSO) = \arg\min_{\mathcal{B}} \mathcal{B}^T \left(\frac{\mathcal{X}^T\mathcal{X} + \lambda_2\mathcal{L}}{1+\lambda_2} \right) \mathcal{B} - 2\mathcal{Y}^T\mathcal{X}\mathcal{B} + \lambda_1|\mathcal{B}|_1. \tag{S2.6}$$

If $\lambda_2 = 0$ in (S2.6) then the SPLASSO simply becomes a series of LASSO models. On the other hand, by setting $\lambda_2 > 0$ we connect each $\boldsymbol{\beta}_i$ to its neighbors. This can be seen more clearly by writing out the first term in (S2.6). Letting $\mathbf{S} = \mathbf{X}^T\mathbf{X}$ we have

$$\frac{\mathcal{X}^T\mathcal{X} + \lambda_2\mathcal{L}}{1+\lambda_2} = \begin{pmatrix} \frac{\mathbf{S}+\lambda_2(d_1-w_{11})\mathbf{I}}{1+\lambda_2} & -\frac{\lambda_2 w_{12}\mathbf{I}}{1+\lambda_2} & \cdots & -\frac{\lambda_2 w_{1n}\mathbf{I}}{1+\lambda_2} \\ -\frac{\lambda_2 w_{21}\mathbf{I}}{1+\lambda_2} & \frac{\mathbf{S}+\lambda_2(d_2-w_{22})\mathbf{I}}{1+\lambda_2} & \cdots & -\frac{\lambda_2 w_{2n}\mathbf{I}}{1+\lambda_2} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{\lambda_2 w_{n1}\mathbf{I}}{1+\lambda_2} & -\frac{\lambda_2 w_{n2}\mathbf{I}}{1+\lambda_2} & \cdots & \frac{\mathbf{S}+\lambda_2(d_n-w_{nn})\mathbf{I}}{1+\lambda_2} \end{pmatrix}. \tag{S2.7}$$

The ij^{th} block in (S2.7) tells us the degree of connectivity between $\boldsymbol{\beta}_i$ and $\boldsymbol{\beta}_j$. Comparing

this to the elastic net, the matrices, $(\mathbf{S} + \lambda_2(d_i - w_{ii})\mathbf{I})/(1+\lambda_2)$, $i = 1, \ldots, n$, along the diagonal are the same, less the scalar term $d_i - w_{ii}$.

The relationship between the SPLASSO and elastic net can be made even clearer if we consider a slight variation on the SPLASSO penalty, specifically if we change the penalty term to be

$$\sum_{ij} \left\| \frac{\boldsymbol{\beta}_i}{\sqrt{d_i}} - \frac{\boldsymbol{\beta}_j}{\sqrt{d_j}} \right\|^2 w_{ij} = \mathcal{B}^T \mathcal{L}_{sym} \mathcal{B}, \tag{S2.8}$$

where $\mathcal{L}_{sym} = \mathbf{I} - \mathcal{D}^{-1/2}\mathcal{W}\mathcal{D}^{-1/2}$. The matrix $\mathcal{L}_{sym}$ is referred to as the "normalized" graph Laplacian in the spectral clustering literature. Letting $\mathcal{W}_{sym} = \mathcal{D}^{-1/2}\mathcal{W}\mathcal{D}^{-1/2}$ we have the following result,

Corollary S2.1. *Given the modified penalty term in (S2.8), data $(\mathcal{Y}, \mathcal{X})$ and regularization parameters (λ_1, λ_2), the SPLASSO estimates $\mathcal{B}_{sym}(SPLASSO)$ are given by*

$$\hat{\mathcal{B}}_{sym} = \arg\min_{\mathcal{B}} \mathcal{B}^T \left(\frac{\mathcal{X}^T\mathcal{X} + \lambda_2\mathbf{I}}{1+\lambda_2} - \frac{\lambda_2 \mathcal{W}_{sym}}{1+\lambda_2} \right) \mathcal{B} - 2\mathcal{Y}^T\mathcal{X}\mathcal{B} + \lambda_1|\mathcal{B}|_1. \tag{S2.9}$$

The matrix $(\mathcal{X}^T\mathcal{X} + \lambda_2\mathbf{I})/(1+\lambda_2)$ in (S2.9) now has the exact same form as the elastic net. This representation of the SPLASSO illustrates the balance between the decorrelation of variables and the constraint of spatial smoothness.

In HSI applications the decorrelation property can be particularly important as end-members are often strongly correlated. It was shown that when two (or more) covariates are exactly correlated the LASSO fails to find a unique solution (Zou & Hastie (2005), Lemma 2) and that it generally encounters difficulties when covariates are highly correlated. One of the key properties of the elastic net is that it is able to overcome this through the introduction of the l_2 penalty term. In the following theorem we show that the SPLASSO shares a similar property.

Theorem S2.2. *Given data $(\mathbf{y}_i, \mathbf{X})$, $i = 1, \ldots, n$ and parameters (λ_1, λ_2), let the response*

$\mathbf{y}_i$ *be centered and the predictors* $\mathbf{X}$ *standardized to have unit norm. With* $\hat{\boldsymbol{\beta}}_i(\lambda_1, \lambda_2)$ *being defined as the naïve SPLASSO estimate, suppose* $\hat{\beta}_{i,l}(\lambda_1, \lambda_2)\hat{\beta}_{i,s}(\lambda_1, \lambda_2) > 0$. *Then with*

$$D_{\lambda_1,\lambda_2}(l, s) = \frac{1}{||\mathbf{y}_i||}|\hat{\beta}_{i,l}(\lambda_1, \lambda_2) - \hat{\beta}_{i,s}(\lambda_1, \lambda_2)|,$$

we have

$$D_{\lambda_1,\lambda_2}(l, s) \leq \frac{1}{\lambda_2}\sqrt{2(1-\rho)} + \frac{1}{||\mathbf{y}_i||}\sum_{j \in N_k(\mathbf{y}_i)} |\beta_{j,l} - \beta_{j,s}|w_{ij}, \tag{S2.10}$$

where $\rho = \mathbf{x}_l^T\mathbf{x}_s$ *is the sample correlation.*

What this theorem tells us is that under the SPLASSO model, when two variables $\mathbf{x}_l$ and $\mathbf{x}_s$ are highly correlated, the differences between their coefficient estimates becomes progressively smaller. Unlike the elastic net however, there is an additional term incorporating the differences associated with neighboring coefficients.

S2.5 Connections with univariate soft thresholding

In this section we take a closer look at the behavior of the SPLASSO model as the weight placed on the spatial regularization term increases to infinity. Using (S2.6) and (S2.9) in Theorem S2.1 and Corollary S2.1 respectively, straightforward calculations show us that the SPLASSO estimate becomes,

$$\arg\min_{\mathcal{B}} \mathcal{B}^T\mathcal{L}\mathcal{B} - 2\mathcal{Y}^T\mathcal{X}\mathcal{B} + \lambda_1|\mathcal{B}|_1, \text{ as } \lambda_2 \to \infty.$$

Taking a closer look at the objective function above, this can be re-expressed as

$$\sum_{i=1}^{n}\left(-2\mathbf{y}_i^T\mathbf{X}\boldsymbol{\beta}_i + \sum_{j \in N_k(\mathbf{y}_i)} ||\boldsymbol{\beta}_i - \boldsymbol{\beta}_j||^2 w_{ij} + \lambda_1|\boldsymbol{\beta}_i|_1\right),$$

with straightforward modifications for $\mathcal{L}_{sym}$. Written this way, the solution for a particular $\beta_{i,l}$, given its neighbors is simply

$$\hat{\beta}_{i,l} = \frac{1}{2}\text{sgn}\left(\mathbf{y}_i^T\mathbf{x}_l + \sum_{j\in N_k(\mathbf{y}_i)} \beta_{j,l}w_{ij}\right)\left(\left|\mathbf{y}_i^T\mathbf{x}_l + \sum_{j\in N_k(\mathbf{y}_i)} \beta_{j,l}w_{ij}\right| - \lambda_1\right)_+,$$

as $\lambda_2 \to \infty$. Thus as $\lambda_2 \to \infty$ the correlation between coefficients is shrunk to zero, leaving $\mathbf{y}_i^T\mathbf{x}_l + \sum_{j\in N_k(\mathbf{y}_i)} \beta_{j,l}w_{ij}$, the univariate regression estimator plus the weighted average of the neighboring coefficients.

S2.6 Rescaling and the SPLASSO

In Zou & Hastie (2005) the authors showed that the introduction of the l_2 penalty term resulted in the elastic net's coefficient estimates receiving shrinkage from both the l_1 and the l_2 penalties. In order to address this, they proposed rescaling the "naïve" solution as they called it by a factor of $1 + \lambda_2$. Their justification for this stemmed primarily from two points; first, that the LASSO is minimax optimal (Donoho et al. (1995)) and second, in the orthonormal setting the naïve elastic net is simply the LASSO solution scaled by $1/(1+\lambda_2)$. Intuitively then, multiplying the naïve elasatic net solution by $(1 + \lambda_2)$ would in yield a minimax optimal solution. This logic was then extended to the general non-orthonormal case.

In the context of the SPLASSO we recommend against this rescaling. To see why, we begin by re-expressing the quantity in (S2.4) as

$$\frac{1}{1+\lambda_2}\text{sgn}(\hat{\beta}_{i,l} + \lambda_2\alpha_{i,l})\left(|\hat{\beta}_{i,l} + \lambda_2\alpha_{i,l}| - \frac{\lambda_1}{2}\right)_+. \tag{S2.11}$$

From (S2.11) we see that there is the same scaling factor of $1/(1 + \lambda_2)$ as in the elastic net. However, this scaling factor plays an important role in balancing the influence of the surrounding coefficient estimates and the OLS estimate. If we remove it this will cause the overall estimate to be inflated by the term $\lambda_2\alpha_{i,l}$.

S3 A computationally efficient solution to the SPLASSO

Even with leveraging parallelization as described in Section4.2, for problems with sufficiently large n the process of solving the LARS-SPLASSO algorithm at each point can still become computationally expensive. In the context of some HSI applications this can be a major issue as images may often have a spatial resolution of more than 1000×1000 and a spectral resolution of 1000 or more. In the following section we propose a more efficient solution to the SPLASSO problem, also based on the ideas of coordinate descent.

In recent work by Friedman et al. (2007) a coordinate descent algorithm was proposed for the LASSO and for several related methods. In that work it was shown that coordinate descent provided considerable improvements in computational speed over competitors (including LARS and other state-of-the-art optimization techniques). Additionally, in Liu et al. (2009) a similar *blockwise* coordinate descent algorithm was proposed for the multivariate response case.

Building on these ideas we show how a similar blockwise-coordinate descent based approach as in Liu et al. (2009) can be implemented for the SPLASSO. Our algorithm consists of simultaneously updating the coefficients for a given $\boldsymbol{\beta}_i$ while holding all the others fixed, then cycling through this process until convergence. Suppose the current estimates are $\hat{\boldsymbol{\beta}}_i$, $i = 1, \ldots, n$. Then $\boldsymbol{\beta}_q$ is updated as

$$\begin{aligned} \hat{\boldsymbol{\beta}}_q = \arg\min_{\hat{\beta}_{q,k}} ||\mathbf{r}_{q,k} - \mathbf{x}_k \hat{\beta}_{q,k}||^2 \qquad & +\lambda_1 |\beta_{q,k}| \\ + \lambda_2 \sum_{j \in N(\mathbf{y}_q)} (\hat{\beta}_{q,k} - \beta_{j,k})^2 w_{qj}, \text{ for } k = 1, \ldots, m, & \end{aligned} \tag{S3.1}$$

where $\mathbf{r}_{q,k} = \mathbf{y}_q - \sum_{l \neq k} \mathbf{x}_l \beta_{q,l}$ denotes the partial residual vector. It can be shown that (S3.1) can be solved in closed form. Letting $b_{q,k} = \mathbf{r}_{q,k}^T \mathbf{x}_k + \lambda_2 \sum_{j \in N(\mathbf{y}_q)} \beta_{j,k} w_{qj}$ we have

$$\hat{\beta}_{q,k} = \frac{\text{sgn}(b_{q,k})(|b_{q,k}| - \lambda_1/2)_+}{\mathbf{x}_k^T \mathbf{x}_k + \lambda_2 \sum_{j \in N(\mathbf{y}_q)} w_{qj}}. \tag{S3.2}$$

What is appealing about (S3.2) is that most quantities can be pre-computed. Specifically for $\mathbf{r}_{q,k}^T\mathbf{x}_k = \mathbf{y}_q^T\mathbf{x}_k - \sum_{l\neq k}\mathbf{x}_l^T\mathbf{x}_k\beta_{q,l}$ the inner products $u_{q,k} = \mathbf{y}_q^T\mathbf{x}_k$, $1 \leq q \leq n$, $1 \leq k \leq m$ and $v_{l,k} = \mathbf{x}_l^T\mathbf{x}_k$, $1 \leq l, k \leq m$ as well as $\sum_{j\in N(\mathbf{y}_q)} w_{qj}$ can all be calculated in advance. This provides considerable savings in computation once we start iterating through the coordinate descent algorithm. Note, this updating procedure we just described is the same *covariance update* discussed in Friedman et al. (2010).

The coordinate descent SPLASSO algorithm begins by initializing $\hat{\boldsymbol{\beta}}_i = 0$, $i = 1, \ldots, n$. To generate the solution path, a decreasing sequence of regularization parameters, $\lambda_1 \in \{C\Delta^t, t = 0, \ldots, t_0\}$, $0 < \Delta < 1$, $t_0 \in \mathbb{Z}^+$, is selected where $C \in \mathbb{R}^+$. Here C is chosen so that for $t = 0$, effectively all the coefficient estimates will be thresholded to 0, and for $t = t_0$ they will be close to the OLS estimates, i.e. no thresholding.

Once the coefficients have been estimated by the coordinate descent algorithm for $t = 0$, the coefficients estimates are recalculated using the previous coefficient estimates as the starting values. This process is repeated for each t. This is the *warm start* concept discussed in Liu et al. (2009) and Friedman et al. (2010). A summary of the algorithm is provided in Figure 4.

Note, additional speedups are possible by indexing those $\hat{\beta}_{q,k}$, which have converged or have been set to 0 for a given λ_1, and skipping them as we iterate through the algorithm.

S4 The Adaptive SPLASSO

One of the shortcomings of the LASSO model is that while the inclusion of the l_1 penalty term comes with the benefit of encouraging sparsity it also introduces biased coefficient estimates (as illustrated in Figure 3 of Samarov et al. (2013)). The adaptive LASSO, or ALASSO (Zou (2006)) was developed to help reduce this bias through the addition of a set of weights for each coefficient $\beta_{i,l}$ on the penalty term λ (or λ_1 in the case of the SPLASSO). The weights for those coefficients whose value is significantly different from 0 would be smaller (thus reducing the amount of shrinkage), and greater for those that were closer to 0 (increasing

Coordinate Descent SPLASSO:

- Compute inner products $u_{q,k}$, $1 \leq q \leq n$, $1 \leq k \leq m$ and $v_{l,k}$, $1 \leq l, k \leq m$ and spatial weight w_{ij}.
- Select value $0 < \Delta < 1$, t_0 and C.
- For each $\lambda_1 \in \{C\Delta^t, t = 1, \ldots, t_0\}$, iterate the through the following until convergence:
  1. If $t = 0$ set as the starting value $\boldsymbol{\beta}_i = 0$, and $\boldsymbol{\beta}_i = \hat{\boldsymbol{\beta}}_i$ if $t > 0$, where $\hat{\boldsymbol{\beta}}_i$ is the previous iterations estimates of the coefficients.
  2. For each $q \in \{1, \ldots, n\}$ and every $k \in \{1, \ldots, m\}$ calculate
  $$\hat{\beta}^0_{q,k} = \frac{\text{sgn}(b_{q,k})(|b_{q,k}| - \lambda_1/2)_+}{\mathbf{x}_k^T \mathbf{x}_k + \lambda_2 \sum_{j \in N(\mathbf{y}_q)} w_{qj}}$$
  3. If $|\hat{\beta}^0_{q,k} - \hat{\beta}_{q,k}| < \epsilon$, for all q and k, ϵ small stop, else repeat.
- Output SPLASSO solution path $\hat{\boldsymbol{\beta}}_i$.

Figure 4: The Coordinate Descent SPLASSO algorithm

the amount of shrinkage). Using a similar framework as the ALASSO, we propose a variant on the SPLASSO we call the adaptive SPLASSO (or ASPLASSO).

As with the ALASSO, we introduce a set of weights for each $\beta_{i,l}$ in our model, say $\phi_{i,l}$, $i = 1, \ldots, n$, $l = 1, \ldots, m$, which are smaller or larger according to the coefficients relative importance. The ASPLASSO model can be written as

$$\sum_{i=1}^{n} ||\mathbf{y}_i - \mathbf{X}\boldsymbol{\beta}_i||^2 + \lambda_1 \sum_{l=1}^{m} \phi_{i,l}|\beta_{i,l}| + \lambda_2 \sum_{j \in N(\mathbf{y}_i)} w_{ij} \sum_{l=1}^{m} (\beta_{i,l} - \beta_{j,l})^2. \tag{S4.1}$$

The weights $\phi_{i,l}$ can take on a number of forms, a reasonable choice is the reciprocal of the weighted average of the neighboring coefficient estimates,

$$\phi_{i,l} = \frac{1}{\sum_{j \in N_k(\mathbf{y})} w_{ij}\beta_{j,l}}. \tag{S4.2}$$

This model has a similar interpretation to the standard SPLASSO model discussed in Section 3 with one slight modification resulting from the introduction of the weights $\phi_{i,l}$. The effect of these weights is most easily seen in the case where we take, $\mathbf{X}$ as orthonormal. With $\hat{b}_{i,l}$, $\alpha_{i,l}$ and γ, as defined in Section 3.2, the ASPLASSO estimate are

$$\hat{\beta}_{i,l}(\text{naïve ASPLASSO}) = \text{sgn}(\hat{b}_{i,l})\left(|\hat{b}_{i,l}| - \frac{\lambda_1}{2}\gamma\phi_{i,l}\right)_+. \tag{S4.3}$$

Setting $\phi_{i,l} = 1/|\hat{b}_{i,l}|$ in (S4.3) we see that as $|\hat{b}_{i,l}|$ increases there will be less weight placed on the penalty term, resulting in a reduction in the amount of shrinkage. On the other hand with $|\hat{b}_{i,l}|$ close to 0 the amount of shrinkage and the likelihood that the estimate will be set to 0 increases.

S4.1 Solving the ASPLASSO

Both the LARS and the coordinate descent approaches used to solve the SPLASSO described in Section 4 of Samarov et al. (2013) can also be used to solve the ASPLASSO. We begin with the LARS formulation. Let $\mathbf{x}^\dagger_{il} = \mathbf{x}_l/\phi_{i,l}$, $\beta^\dagger_{i,l} = \phi_{i,l}\beta_{i,l}$, $w^\dagger_{ijl} = \lambda_2 w_{ij}/\phi^2_{i,l}$ and $\beta^\dagger_{j,l} = \phi_{i,l}\beta_{j,l}$, for $j \in N(\mathbf{y}_i)$. Putting these together, the problem in (S4.1) can then be expressed as

$$\begin{aligned}
&\sum_{i=1}^n ||\mathbf{y}_i - \mathbf{X}\boldsymbol{\beta}_i||^2 + \lambda_1\sum_{l=1}^m \phi_{i,l}|\beta_{i,l}| + \lambda_2 \sum_{j\in N(\mathbf{y}_i)} w_{ij}\sum_{l=1}^m(\beta_{i,l} - \beta_{j,l})^2 \\
&= \sum_{i=1}^n \left\|\mathbf{y}_i - \sum_{l=1}^m \frac{\mathbf{x}_l}{\phi_{i,l}}\phi_{i,l}\beta_{i,l}\right\|^2 + \lambda_1\sum_{l=1}^m \phi_{i,l}|\beta_{i,l}| \\
&\qquad + \lambda_2 \sum_{j\in N(\mathbf{y}_i)}\sum_{l=1}^m \frac{w_{ij}}{\phi^2_{i,l}}(\beta_{i,l} - \beta_{j,l})^2\phi^2_{i,l} \\
&= \sum_{i=1}^n \left\|\mathbf{y}_i - \sum_{l=1}^m \mathbf{x}^\dagger_{il}\beta^\dagger_{i,l}\right\|^2 + \lambda_1\sum_{l=1}^m |\beta^\dagger_{i,l}| + \sum_{j\in N(\mathbf{y}_i)}\sum_{l=1}^m w^\dagger_{ijl}(\beta^\dagger_{i,l} - \beta^\dagger_{j,l})^2 \\
&= \sum_{i=1}^n ||\mathbf{y}_i - \mathbf{X}^\dagger_i\boldsymbol{\beta}^\dagger_i||^2 + \lambda_1|\boldsymbol{\beta}^\dagger_i|_1 + \sum_{j\in N(\mathbf{y}_i)}(\boldsymbol{\beta}^\dagger_i - \boldsymbol{\beta}^\dagger_j)^T\mathbf{W}^\dagger_{ij}(\boldsymbol{\beta}^\dagger_i - \boldsymbol{\beta}^\dagger_j).
\end{aligned} \tag{S4.4}$$

where

$$\mathbf{y}_i^\dagger = \begin{pmatrix} \mathbf{y}_i \\ \mathbf{W}_{i_1}^{\dagger 1/2}\boldsymbol{\beta}_{i_1}^\dagger \\ \vdots \\ \mathbf{W}_{i_{n_k}}^{\dagger 1/2}\boldsymbol{\beta}_{i_{n_k}}^\dagger \end{pmatrix}, \ \mathbf{X}_i^\dagger = (1+\lambda_2)^{-1/2} \begin{pmatrix} \mathbf{X} \\ \mathbf{W}_{i_1}^{\dagger 1/2} \\ \vdots \\ \mathbf{W}_{i_{n_k}}^{\dagger 1/2} \end{pmatrix},$$

$$\boldsymbol{\beta}_i^\dagger = (1+\lambda_2)^{1/2} \begin{pmatrix} \beta_{i,1}^\dagger \\ \vdots \\ \beta_{i,m}^\dagger \end{pmatrix} \mathbf{W}_{ij}^\dagger = \text{diag}(w_{ij1}^\dagger, \ldots, w_{ijm}^\dagger),$$

$$\boldsymbol{\beta}_j^\dagger = \begin{pmatrix} \beta_{j,1}^\dagger \\ \vdots \\ \beta_{j,m}^\dagger \end{pmatrix}, \text{ for } j \in N(\mathbf{y}_i). \tag{S4.5}$$

Through similar calculations to (4.1), and using the quantities in (S4.5), (S4.4) can be written as

$$\sum_{i=1}^{n} ||\mathbf{y}_i^\dagger - \mathbf{X}^\dagger \boldsymbol{\beta}_i^\dagger||^2 + \frac{\lambda_1}{(1+\lambda_2)^{1/2}} |\boldsymbol{\beta}_i^\dagger|_1. \tag{S4.6}$$

The same LARS-SPLASSO algorithm can then be used to solve (S4.6).

The coordinate descent approach to solving the ASPLASSO requires changing the update of $\hat{\beta}_{q,k}^0$ in Step 2 of the coordinate descent SPLASSO algorithm shown in Figure 6 of Samarov et al. (2013) to

$$\hat{\beta}_{q,k}^0 = \frac{\text{sgn}(b_{q,k})(|b_{q,k}| - \phi_{i,l}/2)_+}{\mathbf{x}_k^T \mathbf{x}_k + \lambda_2 \sum_{j \in N(\mathbf{y}_q)} w_{qj}}.$$

Currently work is under way to implement both the LARS and coordinate descent solutions to the ASPLASSO.

S5 Proofs

In this section we provide proofs from some of the results discussed in the previous sections. We begin with Theorem S2.1

Proof. Let $\hat{\mathcal{B}}$ be the solution of (S2.5). Calculations similar to (4.1) show (S2.5) can be written as

$$||\mathcal{Y}^* - \mathcal{X}^*\mathcal{B}^*||^2 + \frac{\lambda_1}{(1+\lambda_2)^{1/2}}|\mathcal{B}^*|_1, \tag{S5.1}$$

where

$$\mathcal{Y}^* = \begin{pmatrix} \mathcal{Y} \\ \mathbf{0} \end{pmatrix}, \ \mathcal{X}^* = \frac{1}{(1+\lambda_2)^{1/2}} \begin{pmatrix} \mathcal{X} \\ \mathcal{L}^{1/2} \end{pmatrix}, \ \mathcal{B}^* = (1+\lambda_2)^{1/2}\mathcal{B}. \tag{S5.2}$$

Using (S5.1) we get

$$\begin{aligned} \hat{\mathcal{B}} &= \arg\min_{\mathcal{B}} \left\| \mathcal{Y}^* - \mathcal{X}^* \frac{\mathcal{B}}{(1+\lambda_2)^{1/2}} \right\|^2 + \frac{\lambda_1}{(1+\lambda_2)^{1/2}} \left| \frac{\mathcal{B}}{(1+\lambda_2)^{1/2}} \right|_1 \\ &= \arg\min_{\mathcal{B}} \mathcal{B}^T \left(\frac{\mathcal{X}^{*T}\mathcal{X}^*}{1+\lambda_2} \right) \mathcal{B} - 2\frac{\mathcal{Y}^{*T}\mathcal{X}^*}{(1+\lambda_2)^{1/2}} + \mathcal{Y}^{*T}\mathcal{Y}^* + \frac{\lambda_1|\mathcal{B}|_1}{1+\lambda_2}. \end{aligned} \tag{S5.3}$$

Plugging in the identities in (S5.2) into (S5.3), we have

$$\begin{aligned} \hat{\mathcal{B}} &= \arg\min_{\mathcal{B}} \frac{1}{1+\lambda_2} \left[\mathcal{B}^T \left(\frac{\mathcal{X}^T\mathcal{X} + \lambda_2\mathcal{L}}{1+\lambda_2} \right) \mathcal{B} - 2\mathcal{Y}^T\mathcal{X}\mathcal{B} + \lambda_1|\mathcal{B}|_1 \right] + \mathcal{Y}^T\mathcal{Y} \\ &= \arg\min_{\mathcal{B}} \mathcal{B}^T \left(\frac{\mathcal{X}^T\mathcal{X} + \lambda_2\mathcal{L}}{1+\lambda_2} \right) \mathcal{B} - 2\mathcal{Y}^T\mathcal{X}\mathcal{B} + \lambda_1|\mathcal{B}|_1. \end{aligned}$$

□

Next we provide a proof of Theorem S2.2. Note the structure of this proof is quite similar to that of Zou & Hastie (2005).

Proof. If $\hat{\beta}_{i,l}(\lambda_1, \lambda_2)\hat{\beta}_{i,s}(\lambda_1, \lambda_2) > 0$, then both $\hat{\beta}_{i,l}(\lambda_1, \lambda_2)$ and $\hat{\beta}_{i,l}(\lambda_1, \lambda_2)$ are non-zero and

have the same sign. Taking the derivative of the SPLASSO loss (3.4) with respect to $\beta_{i,l}$ and $\beta_{i,s}$, and setting equal to zero, one gets

$$\begin{aligned} &-2\mathbf{x}_l^T(\mathbf{y}_i - \mathbf{X}\boldsymbol{\beta}_i(\lambda_1,\lambda_2)) + \lambda_1 \text{sign}(\hat{\beta}_{i,l}(\lambda_1,\lambda_2)) \\ &\quad + 2\lambda_2 \sum_{j\in N_k(\mathbf{y}_i)} (\hat{\beta}_{i,l} - \beta_{j,l}) w_{ij} = 0, \end{aligned} \tag{S5.4}$$

$$\begin{aligned} &-2\mathbf{x}_s^T(\mathbf{y}_i - \mathbf{X}\boldsymbol{\beta}_i(\lambda_1,\lambda_2)) + \lambda_1 \text{sign}(\hat{\beta}_{i,s}(\lambda_1,\lambda_2)) \\ &\quad + 2\lambda_2 \sum_{j\in N_k(\mathbf{y}_i)} (\hat{\beta}_{i,s} - \beta_{j,l}) w_{ij} = 0. \end{aligned} \tag{S5.5}$$

Next we subtract (S5.4) from (S5.5) which gives

$$\begin{aligned} &(\mathbf{x}_s - \mathbf{x}_l)^T(\mathbf{y}_i - \mathbf{X}\hat{\boldsymbol{\beta}}_i(\lambda_1,\lambda_2)) + \lambda_2(\hat{\beta}_{i,l}(\lambda_1,\lambda_2) - \hat{\beta}_{i,s}(\lambda_1,\lambda_2)) - \\ &\lambda_2 \sum_{j\in N_k(\mathbf{y}_i)} (\beta_{j,l} - \beta_{j,s}) w_{ij}, \end{aligned}$$

which is equivalent to

$$\hat{\beta}_{i,l}(\lambda_1,\lambda_2) - \hat{\beta}_{i,s}(\lambda_1,\lambda_2) = \frac{1}{\lambda_2}(\mathbf{x}_l - \mathbf{x}_s)^T \hat{\mathbf{r}}_i(\lambda_1,\lambda_2) + \sum_{j\in N_k(\mathbf{y}_i)} (\beta_{j,l} - \beta_{j,s}) w_{ij}, \tag{S5.6}$$

where $\hat{\mathbf{r}}_i(\lambda_1,\lambda_2) = \mathbf{y}_i - \mathbf{X}\boldsymbol{\beta}_i(\lambda_1,\lambda_2)$ is the residual vector. Since the columns of $\mathbf{X}$ are standardized to have unit norm, $||\mathbf{x}_i - \mathbf{x}_j||^2 = 2(1-\rho)$ where $\rho = \mathbf{x}_i^T\mathbf{x}_j$. Since our loss function is convex,

$$||\hat{\mathbf{r}}_i(\lambda_1,\lambda_2)||^2 + \lambda_2||\hat{\boldsymbol{\beta}}_i(\lambda_1,\lambda_2)||^2 + \lambda_1|\hat{\boldsymbol{\beta}}_i(\lambda_1,\lambda_2)|_1 \leq ||\mathbf{y}_i||^2,$$

so $||\hat{\mathbf{r}}_i(\lambda_1,\lambda_2)|| \leq ||\mathbf{y}_i||^2$. Then (S5.6) implies that

$$D_{\lambda_1,\lambda_2}(l,s) \leq \frac{1}{||\mathbf{y}_i||}\left|\frac{(\mathbf{x}_l - \mathbf{x}_s)^T\hat{\mathbf{r}}_i}{\lambda_2}\right| + \frac{1}{||\mathbf{y}_i||}\left|\sum_{j\in N_k(\mathbf{y}_i)} (\beta_{j,l} - \beta_{j,s}) w_{ij}\right|$$

$$\leq \frac{1}{\lambda_2}\sqrt{2(1-\rho)} + \frac{1}{||\mathbf{y}_i||}\sum_{j\in N_k(\mathbf{y}_i)} |\beta_{j,l} - \beta_{j,s}| w_{ij}$$

□

References

Donoho, D., Johnstone, I., Kerkyacharian, G. & Picard, D. (1995). Wavelet shrinkage: asymptotia (with discussion)? *J.R. Statist. Soc. B* 57 301–369.

Friedman, J., Hastie, T., Hofling, H. & Tibshirani, R. (2007). Pathwise coordinate optimization. *The Annals of Applied Statistics* 1 302–332.

Friedman, J., Hastie, T. & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33 1–22.

Hagen, L. & Kahng, A. (1992). New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. Computer-Aided Design* 11 1074–1085.

Liu, H., Palatucci, M. & Zhang, J. (2009). Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery.

Samarov, D., Litorja, M. & Hwang, J. (2013). The spatial lasso with applications to umixing hyperspectral biomedical images. *Technometrics (submitted)* .

Shi, J. & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 888–905.

von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing* 17 395–416.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association* 101 1418–1429.

Zou, H. & Hastie, T. (2005). Regularization and variable selection via the elastic net. *J Roy Statist Soc Ser B* 67 301–320.