Appendix to

"Automatic Feature Selection via Weighted Kernels and Regularization"

published in the *Journal of Computational and Graphical Statistics*

Genevera I. Allen*

1 Convergence of KNIFE

We will show that the KNIFE algorithm is an approximation to a descent algorithm that under certain conditions on the loss function and kernel, converges to a local minimum of the KNIFE objective function. We also discuss the properties of this descent algorithm and the KNIFE algorithm, showing through numerical examples the converge of the KNIFE algorithm. In special cases, we also show that the KNIFE algorithm converges to a stationary point of the KNIFE objective.

Again, we seek to minimize the following problem:

$$\begin{aligned} \underset{\boldsymbol{\alpha},\mathbf{w}}{\text{minimize}} \quad & f(\boldsymbol{\alpha},\mathbf{w}) = L(\mathbf{Y},\mathbf{K}_{\mathbf{w}}\,\boldsymbol{\alpha}) + \lambda_1\,\boldsymbol{\alpha}^T\,\mathbf{K}_{\mathbf{w}}\,\boldsymbol{\alpha} + \lambda_2 \mathbf{1}^T\,\mathbf{w} \\ \text{subject to} \quad & 0 \leq w_j < 1, \text{ for all } j = 1 \ldots p. \end{aligned}$$

[0]Department of Pediatrics-Neurology, Baylor College of Medicine, Jan and Dan Duncan Neurological Research Institute, Texas Children's Hospital, & Department of Statistics, Rice University, MS 138, 6100 Main St., Houston, TX 77005 (email: gallen@rice.edu)

We repeat the KNIFE algorithm here for completeness:

Algorithm 1 KNIFE Algorithm

1. Initialize $\boldsymbol{\alpha}^{(0)}$ and $\mathbf{w}^{(0)}$ where $0 < w_j^{(0)} < 1$ for $j = 1 \dots p$.
2. Set $\boldsymbol{\alpha}^{(t)} = \text{argmin}_{\boldsymbol{\alpha}} \left\{ L(\mathbf{Y}, \mathbf{K}_{\mathbf{w}^{(t-1)}}\, \boldsymbol{\alpha}) + \lambda_1\, \boldsymbol{\alpha}^T\, \mathbf{K}_{\mathbf{w}^{(t-1)}}\, \boldsymbol{\alpha} \right\}$.
3. Set $\mathbf{w}^{(t)}$ to the solution of:

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}} \;\; & L(\mathbf{Y}, \mathbf{B}\, \boldsymbol{\alpha}^{(t)} + \mathbf{A}\, \mathbf{w}) + \lambda_1 (\boldsymbol{\alpha}^{(t)})^T\, \mathbf{A}\, \mathbf{w} + \lambda_2 \mathbf{1}^T\, \mathbf{w} \\ \text{subject to} \;\; & 0 \leq w_j < 1, j = 1, \dots p, \end{aligned}$$

   where $\mathbf{B} \in \Re^{n \times n} : \; \mathbf{B}_{ii'} = k_{\mathbf{w}^{(t-1)}}(i, i') - \bigtriangledown k_{\mathbf{w}^{(t-1)}}(i, i')^T\, \mathbf{w}^{(t-1)}$, $\mathbf{A} \in \Re^{n \times p} : \; \mathbf{A}_{ii'} = \sum_{i'=1}^{n} \alpha_{i'} \bigtriangledown k_{\mathbf{w}^{(t-1)}}(i, i')^T$, and $\nabla k_{\mathbf{w}^{(t-1)}}(i, i')$ is the gradient of the (i, i') element of $\mathbf{K}_{\mathbf{w}^{(t-1)}}$ with respect to $\mathbf{w}^{(t-1)}$.
4. Repeat steps 2-3 until convergence.

Now, consider the following algorithm:

Algorithm 2 KNIFE Descent Algorithm

1. Initialize $\boldsymbol{\alpha}^{(0)}$ and $\mathbf{w}^{(0)}$ where $0 < w_j^{(0)} < 1$ for $j = 1 \dots p$.
2. Set $\boldsymbol{\alpha}^{(t)} = \text{argmin}_{\boldsymbol{\alpha}} \left\{ L(\mathbf{Y}, \mathbf{K}_{\mathbf{w}^{(t-1)}}\, \boldsymbol{\alpha}) + \lambda_1\, \boldsymbol{\alpha}^T\, \mathbf{K}_{\mathbf{w}^{(t-1)}}\, \boldsymbol{\alpha} \right\}$.
3. Estimate $\mathbf{w}^{(t)}$:
   (a) Define $\tilde{f}(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t)}) = L(\mathbf{Y}, \mathbf{B}\, \boldsymbol{\alpha}^{(t)} + \mathbf{A}\, \mathbf{w}) + \lambda_1 (\boldsymbol{\alpha}^{(t)})^T\, \mathbf{A}\, \mathbf{w} + \lambda_2 \mathbf{1}^T\, \mathbf{w}$ where $\mathbf{B} \in \Re^{n \times n} : \mathbf{B}_{ii'} = k_{\mathbf{w}^{(t-1)}}(i, i') - \bigtriangledown k_{\mathbf{w}^{(t-1)}}(i, i')^T\, \mathbf{w}^{(t-1)}$, $\mathbf{A} \in \Re^{n \times p} : \mathbf{A}_{ii'} = \sum_{i'=1}^{n} \boldsymbol{\alpha}_{i'}^{(t)} \bigtriangledown k_{\mathbf{w}^{(t-1)}}(i, i')^T$, and $\nabla k_{\mathbf{w}^{(t-1)}}(i, i')$ is the gradient of the (i, i') element of $\mathbf{K}_{w^{(t-1)}}$ with respect to $\mathbf{w}^{(t-1)}$.
   (b) Compute a descent direction, $\triangle \mathbf{w}$, for $\tilde{f}(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t)})$ with respect to $\mathbf{w}^{(t)}$ over the set $\{\mathbf{w} : 0 \leq \mathbf{w}_j^{(t)} < 1 \text{ for } j = 1, \dots p\}$.
   (c) Conduct a line search: $s = \text{argmin}_{u \geq 0} f(\boldsymbol{\alpha}, \mathbf{w}^{(t-1)} + u \triangle \mathbf{w})$
   (d) Set $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} + s \triangle \mathbf{w}$.
   (e) Repeat steps (a) - (d) until convergence.
4. Repeat steps 2-3 until convergence.

Proposition 1. *Let the loss function, $L(\mathbf{Y}, \mathbf{K}_{\mathbf{w}}\, \boldsymbol{\alpha})$, be a convex and continuously differentiable function of $\boldsymbol{\alpha}$ and let the weighted kernel, $k_{\mathbf{w}}(\mathbf{x}_i, \mathbf{x}_{i'})$, be a convex or concave and continuously differentiable function of $\mathbf{w}$. Then, Algorithm 2, is a descent algorithm and converges to a local minimum of the objective $f(\boldsymbol{\alpha}, \mathbf{w})$.*

Proof. We will show that estimation with respect to $\boldsymbol{\alpha}$, Step 2, and with respect to $\mathbf{w}$, Step 3, in the KNIFE descent algorithm monotonically decreases the objective $f(\boldsymbol{\alpha}, \mathbf{w})$. Since this objective is bounded below by zero, this ensures convergence. We will also show that the solution to this algorithm is a local minimum, meaning that at the solution $(\boldsymbol{\alpha}^*, \mathbf{w}^*)$, $\nabla_{\boldsymbol{\alpha}} f(\boldsymbol{\alpha}^*, \mathbf{w}^*) = 0$ and $\nabla_{\mathbf{w}} f(\boldsymbol{\alpha}^*, \mathbf{w}^*) = 0$.

It is easy to see that block-wise minimization with respect to $\boldsymbol{\alpha}$, Step 2, solves a convex problem. Therefore Step 2 decreases the objective and satisfies $\nabla_{\boldsymbol{\alpha}} f(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t)}) = 0$ after each step.

Next, we consider estimation with respect to $\mathbf{w}$ in Step 3. It is obvious that if $\nabla_{\mathbf{w}} f(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t-1)}) = 0$, Step 3 returns $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)}$ and this step is non-increasing. Then, we consider the case where $\mathbf{w}^{(t-1)}$ is not an optimal point. Without loss of generality and to avoid notational complexities we can consider minimizing an objective $f(\mathbf{w}) = h(g(\mathbf{w}))$ where $h()$ is a convex and continuously differentiable function of $\mathbf{w}$ and $g()$ is a convex or concave and continuously differentiable function of $\mathbf{w}$. In this notation, if we denote the previous solution as $\mathbf{w}'$, Step 3 computes a descent direction of $\tilde{f}_{\mathbf{w}'}(\mathbf{w}) = h(g(\mathbf{w}') + \nabla g(\mathbf{w}')^T(\mathbf{w} - \mathbf{w}'))$. Applying standard definitions, it is easy to see that since $h()$ and $g()$ are continuously differentiable and hence locally Lipshcitz continuous, that $f()$ is also locally Lipshcitz in $\mathbf{w}$. Furthermore, since $h()$ and $g()$ are convex or concave, there exists an open neighborhood of any $\mathbf{w}'$ such that $\nabla g(\mathbf{w}') \neq 0$ and $\nabla h(\mathbf{w}') \neq 0$ in which $h()$ and $g()$ are monotonic (Bertsekas et al., 2003). Putting these two facts together, we have that there exists an open neighborhood of $\mathbf{w}'$, $\mathcal{N}(\mathbf{w}')$, for all $\mathbf{w}'$ such that $\nabla_{\mathbf{w}} f(\mathbf{w}') \neq 0$ in which $f(\mathbf{w})$ for $\mathbf{w} \in \mathcal{N}(\mathbf{w}')$ is monotonic. We will then restrict our consideration to four cases in this monotonic neighborhood: $f()$ is strictly decreasing or non-decreasing with $g()$ convex or $f()$ is strictly increasing or non-increasing with $g()$ concave.

Let us first consider the strictly decreasing case with $g()$ convex. Letting $\mathbf{w}^*$ be a mini-

mum of $\tilde{f}_{\mathbf{w}'}(\mathbf{w})$ in $\mathcal{N}(\mathbf{w}')$, we have

$$
\begin{aligned}
g(\mathbf{w}) &\geq g(\mathbf{w}') + \nabla g(\mathbf{w}')^T(\mathbf{w} - \mathbf{w}') \quad \forall\, \mathbf{w}, \mathbf{w}' && \because g() \text{ is convex.} \\
h(g(\mathbf{w}^*)) &\leq h\left(g(\mathbf{w}') + \nabla g(\mathbf{w}')^T(\mathbf{w}^* - \mathbf{w}')\right) && \because h() \text{ is strictly decreasing.} \\
f(\mathbf{w}^*) &\leq \tilde{f}_{\mathbf{w}'}(\mathbf{w}^*) \leq \tilde{f}_{\mathbf{w}'}(\mathbf{w}') && \because \text{by definition of } \mathbf{w}^*. \\
&\leq f(\mathbf{w}') && \because \tilde{f}_{\mathbf{w}'}(\mathbf{w}') = f(\mathbf{w}').
\end{aligned}
$$

Therefore, Step 3 decreases the original objective when $f()$ is strictly decreasing at $\mathbf{w}^{(t-1)}$. (Note that the line search ensures that $\mathbf{w}^{(t)}$ must remain in a neighborhood in which $f()$ is strictly decreasing).

Now, when $f(\mathbf{w}')$ is non-decreasing and $g()$ is convex, we have that $f()$ is locally convex (Bertsekas et al., 2003), which can be seen by considering the Hessian of $f()$ (Boyd and Vandenberghe, 2004). Then, taking the descent direction $\triangle\, \mathbf{w}$ to be $\triangle\, \mathbf{w} = -\nabla \tilde{f}_{\mathbf{w}'}(\mathbf{w})$, we have a gradient descent step on $\tilde{f}()$. But, since $\nabla \tilde{f}_{w'}(\mathbf{w}) = \nabla f(\mathbf{w}')$, this also is a gradient descent step on $f()$. Again, the line search ensures that only descent steps are taken and that $\mathbf{w}^{(t)}$ remains in a neighborhood which is non-decreasing. The proof of convergence for Step 3, then follows the convergence analysis of gradient descent algorithms (Boyd and Vandenberghe, 2004).

The argument when $g()$ is concave is analogous. Consider the case when $f()$ is strictly increasing:

$$
\begin{aligned}
g(\mathbf{w}) &\leq g(\mathbf{w}') + \nabla g(\mathbf{w}')^T(\mathbf{w} - \mathbf{w}') \quad \forall\, \mathbf{w}, \mathbf{w}' && \because g() \text{ is concave.} \\
h(g(\mathbf{w}^*)) &\leq h\left(g(\mathbf{w}') + \nabla g(\mathbf{w}')^T(\mathbf{w}^* - \mathbf{w}')\right) && \because h() \text{ is strictly increasing,}
\end{aligned}
$$

and the remainder of the argument is identical to the above case. Similarly, when $f(w')$ is non-increasing and $g()$ is concave, we have that $f()$ is locally convex (Bertsekas et al., 2003). Again, the remainder of the argument is identical to the above case.

Finally, while $\nabla_{\mathbf{w}} f(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t)})$ may not necessarily equal zero after each step, this gradient condition will be satisfied at the solution. This occurs as from the above argument, if $\mathbf{w}^{(t-1)}$ is not an optimal point, then there exists a feasible descent direction and convergence has not been achieved. Therefore, we have shown that estimation of $\mathbf{w}^{(t)}$ in Step 3 necessarily decreases the objective $f(\boldsymbol{\alpha}^{(t)}, \mathbf{w})$, and that Algorithm 2 is a descent algorithm that converges to a local minimum of $f(\boldsymbol{\alpha}, \mathbf{w})$.

□

Before discussing how this descent algorithm relates to the KNIFE algorithm some additional remarks are warranted. First, the exact line search can be replaced by a backtracking or other line searching method. The result and proof remain unchanged as the argument proving convergence of gradient descent methods with the various line searches is employed. Second, notice that the arguments in the proof indicate that for all $\mathbf{w}^{(t-1)}$ that are not optimal points ($\nabla_{\mathbf{w}} f(\boldsymbol{\alpha}^{(t)}, \mathbf{w}^{(t-1)}) = 0$), there exists a neighborhood of $\mathbf{w}^{(t-1)}$ in which minimizing $\tilde{f}_{\mathbf{w}^{(t-1)}}(\boldsymbol{\alpha}, \mathbf{w})$ with respect to $\mathbf{w}$ will decrease the objective $f(\boldsymbol{\alpha}, \mathbf{w})$. In the KNIFE descent algorithm, the line search ensures that each step remains in this neighborhood.

The KNIFE algorithm replaces the descent direction and line search for estimating $\mathbf{w}^{(t)}$ with a full minimization of the linearized objective $\tilde{f}_{\mathbf{w}^{(t-1)}}(\boldsymbol{\alpha}, \mathbf{w})$ with respect to $\mathbf{w}$. Thus, the KNIFE algorithm is an approximation to the descent algorithm, and is not guaranteed to strictly decrease the objective at each iteration. Since there always exists a neighborhood of $\mathbf{w}^{(t-1)}$ in which the objective will decrease, however, one can restrict the range considered either explicitly, $|| \mathbf{w}^{(t-1)} - \mathbf{w}^{(t)} || \leq c$ or implicitly through adding a penalty, $\mu || \mathbf{w}^{(t-1)} - \mathbf{w}^{(t)} ||$ to the objective and find c or μ through a line search. In practice, however, we have observed that these are rarely needed as the KNIFE algorithm almost always strictly decreases the objective at each iteration for common kernels and loss functions. We note that we have opted to present the KNIFE approximation algorithm as the approach is faster than conducting a full line search as in the KNIFE descent algorithm. Our implementation of the KNIFE algorithm them employs the KNIFE approximation, checking the objective at each

step to ensure it is decremented. If this is not the case, a line search as in the KNIFE descent algorithm is employed.

For certain loss functions and kernels, however, we can obtain stronger convergence results for both the KNIFE algorithm and KNIFE descent algorithm.

Proposition 2. *If the KNIFE algorithm or KNIFE descent algorithm finds a unique minimum for the coefficients, $\boldsymbol{\alpha}$, and the weights, $\mathbf{w}$, in each step and the loss function and kernel are continuously differentiable, then the algorithm monotonically decreases the objective and converges to a stationary point of $f(\boldsymbol{\alpha}, \mathbf{w})$.*

Proof. Differentiability of the loss function and kernel implies that $f(\boldsymbol{\alpha}, \mathbf{w})$ is regular on its domain. This along with unique minima in both blocks of coordinates satisfies conditions for monotonic convergence to a stationary point for non-convex functions. Differentiability can be relaxed to weaker conditions for regularity (Tseng, 2001). □

The conditions of Proposition 2 require that the objective be strictly convex in both $\boldsymbol{\alpha}$ and $\mathbf{w}$. One such example is the squared error loss with linear kernel. We have mentioned that these examples are bi-convex, and thus the KNIFE algorithm simply iterates between minimization with respect to the coefficients and then the feature weights. As discussed in the manuscript, these can also be written as a convex problem.

While we have discussed convergence properties of the KNIFE algorithm and descent algorithm, the solution will depend on the starting values of $\mathbf{w}$. Even if the conditions of Proposition 2 are satisfied, non-convex functions have potentially many stationary points. Thus, we recommend initializing the KNIFE algorithm at several random starting points and taking the solution which gives the minimum objective value. We investigate this as well as the convergence of the KNIFE approximation algorithm in a small numerical example in the left panel of Figure 1. Here, the KNIFE objective is shown for several iterations of the algorithm starting from random weight initializations. We see that the approximation strictly decreases the objective in this example.

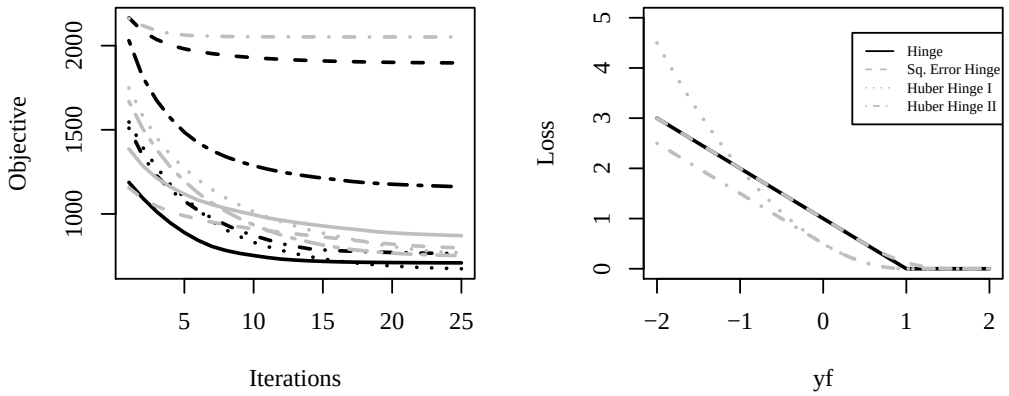

Figure 1: *(left) KNIFE objective for iterations of the KNIFE algorithm starting at 10 random sets of weights. Here, a radial kernel with squared error loss is used. (right) Smooth approximations to the non-differentiable hinge loss for the support vector machine.*

Notice that both of the convergence results require the loss function to be continuously differentiable. While many loss functions such as squared error and binomial deviance are smooth, there is one notable exception, namely the hinge loss of support vector machines. Without smoothness conditions on the loss function, there may not be a feasible descent direction in Step 3 (b) of Algorithm 2 that decreases the original objective. Thus, the coordinate-wise minimizations of KNIFE for SVMs may never converge. Hence, we employ smooth approximations to the non-differentiable hinge loss such as squared error hinge and a Huberized hinge loss (Wang et al., 2008). These are shown in the right panel of Figure 1. Throughout this paper, KNIFE for SVMs is used with one of these smooth loss functions. Additionally, we note that in our experiments and examples, this approximation to the hinge loss generally decreases the original SVM objective function.

References

Bertsekas, D., A. Nedi, and A. Ozdaglar (2003). *Convex analysis and optimization*. Athena Scientific.

Boyd, S. and L. Vandenberghe (2004). *Convex optimization*. Cambridge Univ Pr.

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal Optimization Theory and Applications 109*(3), 475–494.

Wang, L., J. Zhu, and H. Zou (2008). Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics 24*(3), 412–419.