FEATURE-WEIGHTED ELASTIC NET: USING FEATURES OF FEATURES" FOR BETTER PREDICTION

J. Kenneth Tay[1], Nima Aghaeepour[2,3,4], Trevor Hastie[1,4]
and Robert Tibshirani[1,4]

[1]*Department of Statistics, Stanford University*

[2]*Department of Anesthesiology, Pain, and Perioperative Medicine, Stanford University*

[3]*Department of Pediatrics, Stanford University*

[4]*Department of Biomedical Data Sciences, Stanford University*

Supplementary Material

The online supplementary materials provide (i) details on an alternative algorithm with θ as a parameter, (ii) the proof for Theorem 1, (iii) details on the simulation study in Section 5, and (iv) details on the simulation study in Section 7.

S1 Alternative algorithm with θ as a parameter

Assume that $\mathbf{y}$ and the columns of $\mathbf{X}$ are centered so that $\hat{\beta}_0 = 0$ and we can ignore the intercept term in the rest of the discussion. If we consider θ as an argument of the objective function, then we wish to solve

$$\begin{aligned}(\hat{\beta}, \hat{\theta}) &= \underset{\beta,\theta}{\text{argmin}} \;\; J_{\lambda,\alpha}(\beta, \theta) \\ &= \underset{\beta,\theta}{\text{argmin}} \;\; \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^{p} w_j(\theta) \left[\alpha|\beta_j| + \frac{1-\alpha}{2}\beta_j^2 \right].\end{aligned}$$

J is not jointly convex β and θ, so reaching a global minimum is a difficult task. Instead, we content ourselves with reaching a local minimum. A reasonable approach for doing so is to alternate between optimizing β and θ: the steps are outlined in Algorithm 2.

Unfortunately, Algorithm 2 is slow due to repeated solving of the elastic net problem in Step 2(b)ii for each λ_i. The algorithm does not take advantage of the fact that once α and θ are fixed, the elastic net problem can be solved quickly for an entire path of λ values. We have also found that Algorithm 2 does not predict as well as Algorithm 1 in our simulations.

S2 Proof of Theorem 1

For the moment, consider the more general penalty factor $w_j(\theta) = \dfrac{\sum_{\ell=1}^{p} f(\mathbf{z}_\ell^T \theta)}{p f(\mathbf{z}_j^T \theta)}$, where f is some function with range $[0, +\infty)$. (Fwelnet makes the choice

Algorithm 2 *Minimizing the fwelnet objective function via alternating minimization*

1. Select a value of $\alpha \in [0, 1]$ and a sequence of λ values $\lambda_1 > \ldots > \lambda_m$.
2. For $i = 1, \ldots, m$:
   (a) Initialize $\beta^{(0)}(\lambda_i)$ at the elastic net solution for λ_i. Initialize $\theta^{(0)} = \mathbf{0}$.
   (b) For $k = 0, 1, \ldots$ until convergence:
      i. Fix $\beta = \beta^{(k)}$, update $\theta^{(k+1)}$ via gradient descent. That is, set $\Delta\theta = \left.\dfrac{\partial J_{\lambda_i,\alpha}}{\partial \theta}\right|_{\beta=\beta^{(k)},\theta=\theta^{(k)}}$ and update $\theta^{(k+1)} = \theta^{(k)} - \eta\Delta\theta$, where η is the step size computed via backtracking line search to ensure that $J_{\lambda_i,\alpha}\left(\beta^{(k)}, \theta^{(k+1)}\right) < J_{\lambda_i,\alpha}\left(\beta^{(k)}, \theta^{(k)}\right)$.
      ii. Fix $\theta = \theta^{(k+1)}$, update $\beta^{(k+1)}$ by solving the elastic net with updated penalty factors $w_j(\theta^{(k+1)})$.

$f(x) = e^x$.)

First note that if feature j belongs to group k, then $\mathbf{z}_j^T\theta = \theta_k$, and its penalty factor is

$$w_j(\theta) = \frac{\sum_{\ell=1}^p f(\mathbf{z}_\ell^T \theta)}{pf(\mathbf{z}_j^T\theta)} = \frac{\sum_{\ell=1}^p f(\theta_\ell)}{pf(\theta_k)} = \frac{\sum_{\ell=1}^K p_\ell f(\theta_\ell)}{pf(\theta_k)},$$

where p_ℓ denotes the number of features in group ℓ. Letting $v_k = \frac{f(\theta_k)}{\sum_{\ell=1}^K p_\ell f(\theta_\ell)}$ for $k = 1, \dots, K$, minimizing the fwelnet objective function (3.2) over β and θ reduces to

$$\underset{\beta,\theta}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \frac{\lambda}{p}\sum_{k=1}^K \frac{1}{v_k}\left[\alpha \left\|\beta^{(k)}\right\|_1 + \frac{1-\alpha}{2}\left\|\beta^{(k)}\right\|_2^2\right].$$

For fixed β, we can explicitly determine the v_k values which minimize the expression above. By the Cauchy-Schwarz inequality,

$$\begin{aligned}
&\frac{\lambda}{p}\sum_{k=1}^K \frac{1}{v_k}\left[\alpha \left\|\beta^{(k)}\right\|_1 + \frac{1-\alpha}{2}\left\|\beta^{(k)}\right\|_2^2\right] \\
&= \frac{\lambda}{p}\left(\sum_{k=1}^K \frac{1}{v_k}\left[\alpha \left\|\beta^{(k)}\right\|_1 + \frac{1-\alpha}{2}\left\|\beta^{(k)}\right\|_2^2\right]\right)\left(\sum_{k=1}^K p_k v_k\right) \\
&\geq \frac{\lambda}{p}\left(\sum_{k=1}^K \sqrt{p_k\left[\alpha \|\beta^{(k)}\|_1 + \frac{1-\alpha}{2}\|\beta^{(k)}\|_2^2\right]}\right)^2. \qquad \text{(S2.1)}
\end{aligned}$$

Note that equality is attainable for (S2.1): letting $a_k = \sqrt{\frac{\left[\alpha\left\|\beta^{(k)}\right\|_1 + \frac{1-\alpha}{2}\left\|\beta^{(k)}\right\|_2^2\right]}{p_k}}$, equality occurs when there is some $c \in \mathbb{R}$ such that

$$\begin{aligned}
c \cdot \frac{1}{v_k}\left[\alpha \left\|\beta^{(k)}\right\|_1 + \frac{1-\alpha}{2}\left\|\beta^{(k)}\right\|_2^2\right] &= p_k v_k \qquad \text{for all } k, \\
v_k &= \sqrt{c}\, a_k \qquad \text{for all } k.
\end{aligned}$$

Since $\sum_{k=1}^K p_k v_k = 1$, we have $\sqrt{c} = \dfrac{1}{\sum_{k=1}^K p_k a_k}$, giving $v_k = \dfrac{a_k}{\sum_{k=1}^K p_k a_k}$ for all k. A solution for this is $f(\theta_k) = a_k$ for all k, which is feasible for f having range $[0, \infty)$. (Note that if f only has range $(0, \infty)$, the connection still holds if $\lim_{x \to -\infty} f(x) = 0$ or $\lim_{x \to +\infty} f(x) = 0$: the solution will just have $\theta = +\infty$ or $\theta = -\infty$.)

Thus, the fwelnet solution is

$$\underset{\beta}{\operatorname{argmin}} \quad \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \frac{\lambda}{p} \left(\sum_{k=1}^K \sqrt{p_k \left[\alpha \|\beta^{(k)}\|_1 + \frac{1-\alpha}{2} \|\beta^{(k)}\|_2^2 \right]} \right)^2. \tag{S2.2}$$

When $\alpha = 0$, the penalty term is convex. Writing in constrained form, (S2.2) becomes minimizing $\frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2$ subject to

$$\left(\sum_{k=1}^K \sqrt{p_k} \left\|\beta^{(k)}\right\|_2 \right)^2 \le C \text{ for some constant } C,$$
$$\sum_{k=1}^K \sqrt{p_k} \left\|\beta^{(k)}\right\|_2 \le \sqrt{C}.$$

Converting back to Lagrange form again, there is some $\lambda' \ge 0$ such that the fwelnet solution is

$$\underset{\beta}{\operatorname{argmin}} \quad \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda' \sum_{k=1}^K \sqrt{p_k} \left\|\beta^{(k)}\right\|_2.$$

S3 Details on simulation study in Section 5

S3.1 Setting 1: Noisy version of the true β

1. Set $n = 100$, $p = 50$, $\beta \in \mathbb{R}^{50}$ with $\beta_j = 2$ for $j = 1, \ldots, 5$, $\beta_j = -1$ for $j = 6, \ldots, 10$, and $\beta_j = 0$ otherwise.

2. Generate $x_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, p$.

3. For each $SNR_y \in \{0.5, 1, 2\}$ and $SNR_Z \in \{0.5, 2, 10\}$:

   (a) Compute $\sigma_y^2 = \left(\sum_{j=1}^p \beta_j^2\right) / SNR_y$.

   (b) Generate $y_i = \sum_{j=1}^p x_{ij}\beta_j + \varepsilon_i$, where $\varepsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_y^2)$ for $i = 1, \ldots, n$.

   (c) Compute $\sigma_Z^2 = \text{Var}(|\beta|)/SNR_Z$.

   (d) Generate $z_j = |\beta_j| + \eta_j$, where $\eta_j \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_Z^2)$. Treat this as a column matrix to get $\mathbf{Z} \in \mathbb{R}^{p \times 1}$.

S3.2 Setting 2: Grouped data setting

1. Set $n = 100$, $p = 150$.

2. For $j = 1, \ldots, p$ and $k = 1, \ldots 15$, set $z_{jk} = 1$ if $10(k - 1) < j \leq 10k$, $z_{jk} = 0$ otherwise.

3. Generate $\beta \in \mathbb{R}^{150}$ with $\beta_j = 3$ or $\beta_j = -3$ with equal probability for $j = 1, \ldots, 10G$, $\beta_j = 0$ otherwise. $G = 1$ for the first scenario where the response depends on the first group only, and $G = 4$ for the second scenario where it depends on the first 4 groups.

4. Generate $x_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, p$.

5. For each $SNR_y \in \{0.5, 1, 2\}$:

   (a) Compute $\sigma_y^2 = \left(\sum_{j=1}^{p} \beta_j^2\right) / SNR_y$.

   (b) Generate $y_i = \sum_{j=1}^{p} x_{ij}\beta_j + \varepsilon_i$, where $\varepsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_y^2)$ for $i = 1, \ldots, n$.

S3.3 Setting 3: Noise variables

1. Set $n = 100$, $p = 100$, $\beta \in \mathbb{R}^{100}$ with $\beta_j = 2$ for $j = 1, \ldots, 10$, and $\beta_j = 0$ otherwise.

2. Generate $x_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, p$.

3. For each $SNR_y \in \{0.5, 1, 2\}$:

   (a) Compute $\sigma_y^2 = \left(\sum_{j=1}^{p} \beta_j^2\right) / SNR_y$.

   (b) Generate $y_i = \sum_{j=1}^{p} x_{ij}\beta_j + \varepsilon_i$, where $\varepsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_y^2)$ for $i = 1, \ldots, n$.

(c) Generate $z_{jk} \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$ for $j = 1, \dots, p$ and $k = 1, \dots 10$. Append a column of ones to get $\mathbf{Z} \in \mathbb{R}^{p \times 11}$.

S4 Details on simulation study in Section 7

1. Set $n = 150$, $p = 50$.

2. Generate $\beta_1 \in \mathbb{R}^{50}$ with

$$\beta_{1,j} = \begin{cases} 5 \text{ or } -5 \text{ with equal probability} & \text{for } j = 1, \dots, 5, \\ 2 \text{ or } -2 \text{ with equal probability} & \text{for } j = 6, \dots, 10, \\ 0 & \text{otherwise.} \end{cases}$$

3. Generate $\beta_2 \in \mathbb{R}^{50}$ with

$$\beta_{2,j} = \begin{cases} 5 \text{ or } -5 \text{ with equal probability} & \text{for } j = 1, \dots, 5, \\ 2 \text{ or } -2 \text{ with equal probability} & \text{for } j = 11, \dots, 15, \\ 0 & \text{otherwise.} \end{cases}$$

4. Generate $x_{ij} \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$ for $i = 1, \dots, n$ and $j = 1, \dots, p$.

5. Generate response 1, $\mathbf{y}_1 \in \mathbb{R}^{150}$, in the following way:

   (a) Compute $\sigma_1^2 = \left(\sum_{j=1}^p \beta_{1,j}^2 \right) / 0.5$.

   (b) Generate $y_{1,i} = \sum_{j=1}^p x_{ij}\beta_{1,j} + \varepsilon_{1,i}$, where $\varepsilon_{1,i} \overset{i.i.d.}{\sim} \mathcal{N}\left(0, \sigma_1^2\right)$ for $i = 1, \dots, n$.

6. Generate response 2, $\mathbf{y}_2 \in \mathbb{R}^{150}$, in the following way:

   (a) Compute $\sigma_2^2 = \left(\sum_{j=1}^{p} \beta_{2,j}^2\right)/1.5$.

   (b) Generate $y_{2,i} = \sum_{j=1}^{p} x_{ij}\beta_{2,j} + \varepsilon_{2,i}$, where $\varepsilon_{2,i} \overset{i.i.d.}{\sim} \mathcal{N}\left(0, \sigma_2^2\right)$ for $i = 1, \ldots, n$.