Algorithm EG\(_{\pm}\)(U, (s\(^+\), s\(^-\)), \(\eta\))

**Parameters:**

- \(L\): a loss function from \(\mathbb{R} \times \mathbb{R}\) to \([0, \infty)\),
- \(U\): the total weight of the weight vectors,
- \(s^+\) and \(s^-\): a pair of start vectors in \([0, 1]^N\), with \(\sum_{i=1}^{N} (s^+_i + s^-_i) = 1\), and
- \(\eta\): a learning rate in \([0, \infty)\).

**Initialization:** Before the first trial, set \(w^+_1 = Us^+\) and \(w^-_1 = Us^-\).

**Prediction:** Upon receiving the \(t\)th instance \(x_t\), give the prediction

\[
\hat{y}_t = (w^+_t - w^-_t) \cdot x_t,
\]

**Update:** Upon receiving the \(t\)th outcome \(y_t\), update the weights according to the rules

\[
w^+_{t+1} = U / \sum_{i=1}^{N} (w^+_i r^+_i + w^-_i r^-_i),
\]

\[
w^-_{t+1} = U / \sum_{i=1}^{N} (w^+_i r^+_i + w^-_i r^-_i),
\]

where

\[
r^+_t = \exp(-\eta L_t(\hat{y}_t, s^+_t))
\]

\[
r^-_t = \exp(\eta L_t(\hat{y}_t, s^-_t)) = \frac{1}{r^+_t}.
\]

**FIG. 3.** Exponential gradient algorithm with positive and negative weights \(\text{EG}\_{\pm}(U, (s^+, s^-), \eta)\).

Therefore, \(w^+_t - w^-_t\) is the weight vector which can contain negative components. Further, by using the scaling factor \(U\), we can make the weight vector \(w^+_t - w^-_t\) range over all vectors \(w \in \mathbb{R}\) for which \(\|w\|_1 \leq U\). Although \(\|w^+_t\|_1 + \|w^-_t\|_1\) is always exactly \(U\), vectors \(w^+_t - w^-_t\) with \(\|w^+_t - w^-_t\|_1 < U\) result simply from having both \(w^+_t > 0\) and \(w^-_t > 0\) for some \(t\). For other examples of reductions of this type, see Littlestone et al. (1995).

The parameters of \(\text{EG}\_{\pm}\) are a loss function \(L\), a scaling factor \(U\), a pair \((s^+, s^-)\) of start vectors in \([0, 1]^N\) with \(\sum_{i=1}^{N} (s^+_i + s^-_i) = 1\), and a learning rate \(\eta\). We simply write \(\text{EG}\_{\pm}\) for \(\text{EG}\_{\pm}(U, (s^+, s^-), \eta)\) where \(L\) is the square loss function. As the start vectors for \(\text{EG}\_{\pm}\), one would typically use \((s^+ = s^- = (1/(2N), ..., 1/(2N)))\). This gives \(w^+_1 - w^-_1 = 0\). A typical learning rate function could be \(\eta = 1/(3U^2X^2)\) where \(X\) is an estimated upper bound for the maximum \(L_{\infty}\) norm \(\max_{j} \|x_j\|_{\infty}\) of the instances.

More detailed theoretical results are given in Theorem 5.11.

Again, we introduce one particular variable learning rate version of \(\text{EG}\_{\pm}\). We use the name \(\text{EGV}\_{\pm}\) for the algorithm that is as \(\text{EG}\_{\pm}\) except that (3.10) and (3.11) are replaced by