Algorithm **AdaBoost** (Schapire-Singer variant)

**Assume** $B$ contains functions $b$ taking values in $[-1, 1]$ or $\{\pm 1\}$

**Input** $M$, labeled training set $\mathcal{D}$

**Initialize** $f = 0$

$$w_i^1 = \frac{1}{N} \text{ weight of datapoint } x_i$$

for $k = 1, 2, \ldots M$

1. “learn classifier for $\mathcal{D}$ with weights $w^k$” $\Rightarrow b^k$

2. compute error $\epsilon^k = \sum_{i=1}^{N} w_i^k \frac{1-y_i b^k(x_i)}{2}$

3. set $\beta^k = \frac{1}{2} \ln \frac{1-\epsilon^k}{\epsilon^k}$

4. compute new weights $w_i^{k+1} = \frac{1}{Z_k} w_i^k e^{-\beta^k y_i b^k(x_i)}$ where $Z^k$ is the normalization constant that makes $\sum_i w_i^{k+1} = 1$

**Output** $f(x) = \sum_{k=1}^{M} \beta^k b^k(x)$