Algorithm 1 Gradient Boost

1: $F_0(x) = argmin_\rho \sum_{i=1}^{N} \mathcal{L}(y_i, \rho)$

2: **for** $m = 1$ to M **do**

3: $\quad \tilde{y}_i = -\left[\frac{\partial \mathcal{L}(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, i \in [1, N]$

4: $\quad a_m = argmin_a \sum_{i \in \Omega} \mathcal{L}(\tilde{y}_i, h(x_i, a))$

Some text in between them

Algorithm 2 Blinkered Gradient Boost

$F_0(x) = argmin_\rho \sum_{i=1}^{N} \mathcal{L}(y_i, \rho)$
for $m = 1$ to M **do**
    $B_m = \mathcal{B} - (m-1)\delta$
    $\Omega = \{i \in [1, N] \mid \mathcal{L}(y_i, F_{m-1}(x_i)) < B_m\}$
    $\tilde{y}_i = -\left[\frac{\partial \mathcal{L}(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, i \in \Omega$
    $a_m = argmin_a \sum_{i \in \Omega} \mathcal{L}(\tilde{y}_i, h(x_i, a))$
procedure MyProcedure
    stringlen ← length of *string*
    $i \leftarrow$ *patlen*
top:
    if $i >$ *stringlen* **then return** false
    $j \leftarrow$ *patlen*
loop:
    if *string*(i) = *path*(j) **then**
        $j \leftarrow j - 1$.
        $i \leftarrow i - 1$.
        goto *loop*.
        close;
    $i \leftarrow i + \max(delta_1(string(i)), delta_2(j))$.
    goto *top*.