

## Conformal Bidirectional Floating Search (CBFS): Pseudocode

---

### Algorithm 1: CBFS (main procedure)

---

**Input:** Feature indices  $\{1, \dots, d\}$ ; target subset size  $k$ ; Max Iterations  $I_{\max}$ ; Max Patience  $P_{\max}$ .  
**Output:** Best subset  $S_{\text{best}}$  and best score  $m_{\text{best}} = \mathcal{M}(S_{\text{best}})$ .

```

1  $S \leftarrow \emptyset$ ;  $U \leftarrow \{1, \dots, d\}$ ;  $U^* \leftarrow \emptyset$ ;
2 Until  $|S| = k$  do::
3    $(S, U, U^*) \leftarrow \text{BIDIRECTIONALSEARCH}(S, U, U^*, k)$ ;
4    $U \leftarrow (U \cup U^*) \setminus S$ ;  $U^* \leftarrow \emptyset$ ;
5  $m \leftarrow \mathcal{M}(S)$ ;  $S_{\text{best}} \leftarrow S$ ;  $m_{\text{best}} \leftarrow m$ ;
6  $\text{patience} \leftarrow 0$ ;  $\text{iter} \leftarrow 0$ ;
7 while  $\text{iter} < I_{\max}$  and  $\text{patience} < P_{\max}$  do
8    $\text{iter} \leftarrow \text{iter} + 1$ ;
9    $(S, U, m, S_{\text{best}}, m_{\text{best}}, \text{patience}) \leftarrow$ 
      $\text{REFINESTEP}(S, U, m, S_{\text{best}}, m_{\text{best}}, \text{patience})$ ;
10 return  $S_{\text{best}}, m_{\text{best}}$ ;

```

---



---

### Algorithm 2: BIDIRECTIONALSEARCH

---

**Input:** Current  $S$ , pools  $U$  and  $U^*$ , target size  $k$ .  
**Output:** Updated  $(S, U, U^*)$  with  $|S| = k$ .

```

1 while  $|S| < k$  do
2    $R_{\text{add}} \leftarrow \text{RANKJMI}(U, S)$ ;  $f_{\text{add}} \leftarrow \text{top}(R_{\text{add}})$ ;
3    $S \leftarrow S \cup \{f_{\text{add}}\}$ ;  $U \leftarrow U \setminus \{f_{\text{add}}\}$ ;
4   if  $U \neq \emptyset$  then
5      $R_{\text{remove}} \leftarrow \text{RANKCRFE}(U)$ ;  $f_{\text{prune}} \leftarrow \text{top}(R_{\text{remove}})$ ;
6      $U \leftarrow U \setminus \{f_{\text{prune}}\}$ ;  $U^* \leftarrow U^* \cup \{f_{\text{prune}}\}$ ;
7 return  $(S, U, U^*)$ ;

```

---

---

**Algorithm 3:** REFINESTEP (floating adjustment, one iteration)

---

**Input:**  $S, U$ , current score  $m = \mathcal{M}(S)$ ; global best  $(S_{\text{best}}, m_{\text{best}})$ ; *patience*.

**Output:** Updated  $(S, U, m, S_{\text{best}}, m_{\text{best}}, \textit{patience})$ .

```
1  $R_{\text{remove}} \leftarrow \text{RANKCRFE}(S); \quad f_{\text{remove}} \leftarrow \text{top}(R_{\text{remove}});$ 
2  $R_{\text{add}} \leftarrow \text{RANKJMI}(U, S); \quad f_{\text{add}} \leftarrow \text{top}(R_{\text{add}});$ 
3  $S_{\text{minus}} \leftarrow S \setminus \{f_{\text{remove}}\};$ 
4  $S_{\text{plus}} \leftarrow S \cup \{f_{\text{add}}\};$ 
5  $S_{\text{swap}} \leftarrow (S \setminus \{f_{\text{remove}}\}) \cup \{f_{\text{add}}\};$ 
6  $m_{\text{minus}} \leftarrow \mathcal{M}(S_{\text{minus}}); \quad m_{\text{plus}} \leftarrow \mathcal{M}(S_{\text{plus}}); \quad m_{\text{swap}} \leftarrow \mathcal{M}(S_{\text{swap}});$ 
7  $(S_{\text{new}}, m_{\text{new}}) \leftarrow$ 
    $\arg \min\{(S, m), (S_{\text{minus}}, m_{\text{minus}}), (S_{\text{plus}}, m_{\text{plus}}), (S_{\text{swap}}, m_{\text{swap}})\};$ 
8 if  $m_{\text{new}} < m$  then
9    $S \leftarrow S_{\text{new}}; \quad m \leftarrow m_{\text{new}}; \quad U \leftarrow \{1, \dots, d\} \setminus S;$ 
10   $\textit{patience} \leftarrow 0;$ 
11 else
12   $\textit{patience} \leftarrow \textit{patience} + 1;$ 
13 if  $m < m_{\text{best}}$  then
14   $S_{\text{best}} \leftarrow S; \quad m_{\text{best}} \leftarrow m;$ 
15 return  $(S, U, m, S_{\text{best}}, m_{\text{best}}, \textit{patience});$ 
```

---