Greedy Approximations: $k$-Center

Algorithm Design & Analysis

Spring 2018
Outline
**k-Center Problem**

**Input:** $S = \{s_1, \ldots, s_n\} \subset \mathbb{R}^2$ and an integer $k$

**Problem:** Compute set $C$ of $k$ points in $\mathbb{R}^2$ that minimizes $\max_{s \in S}\{\text{dist}(s, C)\}$

- $\text{dist}(s, C) = \min_{c \in C}\{|s - c|\}$
- $C$ is an $r$-cover of $S$ if $\text{dist}(s, C) \leq r$ for all $s \in S$.
- Given $C$, $r = \max_{s \in S}\{\text{dist}(s, C)\}$ is called the covering radius of $C$.
- Given $k$, we want a $C$ of size $k$ with smallest possible covering radius.
- Infinite search space!
**k-Center Problem**

**Input:** \( S = \{s_1, \ldots, s_n\} \subset \mathbb{R}^2 \) and an integer \( k \)

**Problem:** Compute set \( C \) of \( k \) points in \( \mathbb{R}^2 \) that minimizes \( \max_{s \in S} \{\text{dist}(s, C)\} \)

- \( \text{dist}(s, C) = \min_{c \in C} \{|s - c|\} \)
**k-Center Problem**

**Input:** \( S = \{s_1, \ldots, s_n\} \subset \mathbb{R}^2 \) and an integer \( k \)

**Problem:** Compute set \( C \) of \( k \) points in \( \mathbb{R}^2 \) that minimizes \( \max_{s \in S} \{\text{dist}(s, C)\} \)

- \( \text{dist}(s, C) = \min_{c \in C} \{|s - c|\} \)
- \( C \) is an \( r \)-cover of \( S \) if \( \text{dist}(s, C) \leq r \) for all \( s \in S \).
**k-Center Problem**

**Input:** $S = \{s_1, \ldots, s_n\} \subset \mathbb{R}^2$ and an integer $k$

**Problem:** Compute set $C$ of $k$ points in $\mathbb{R}^2$ that minimizes $\max_{s \in S} \{\text{dist}(s, C)\}$

- $\text{dist}(s, C) = \min_{c \in C}\{|s - c|\}$
- $C$ is an $r$-cover of $S$ if $\text{dist}(s, C) \leq r$ for all $s \in S$.
- Given $C$, $r = \max_{s \in S}\{\text{dist}(s, C)\}$ is called the covering radius of $C$
**k-Center Problem**

**Input:** \( S = \{ s_1, \ldots, s_n \} \subset \mathbb{R}^2 \) and an integer \( k \)

**Problem:** Compute set \( C \) of \( k \) points in \( \mathbb{R}^2 \) that minimizes \( \max_{s \in S} \{ \text{dist}(s, C) \} \)

- \( \text{dist}(s, C) = \min_{c \in C} \{ |s - c| \} \)
- \( C \) is an \( r \)-cover of \( S \) if \( \text{dist}(s, C) \leq r \) for all \( s \in S \).
- Given \( C \), \( r = \max_{s \in S} \{ \text{dist}(s, C) \} \) is called the **covering radius** of \( C \)
- Given \( k \), we want a \( C \) of size \( k \) with smallest possible covering radius
**k-Center Problem**

**Input:** \( S = \{s_1, \ldots, s_n\} \subset \mathbb{R}^2 \) and an integer \( k \)

**Problem:** Compute set \( C \) of \( k \) points in \( \mathbb{R}^2 \) that minimizes \( \max_{s \in S}\{\text{dist}(s, C)\} \)

- \( \text{dist}(s, C) = \min_{c \in C}\{|s - c|\} \)
- \( C \) is an \( r \)-cover of \( S \) if \( \text{dist}(s, C) \leq r \) for all \( s \in S \).
- Given \( C \), \( r = \max_{s \in S}\{\text{dist}(s, C)\} \) is called the *covering radius* of \( C \)
- Given \( k \), we want a \( C \) of size \( k \) with smallest possible covering radius
- Infinite search space!
Bootstrapping to a Good Algorithm

Use the covering radius $r^*$ and a cover $C^*$ of size $k$ with $r(C^*) = r^*$ to find a good approximation that uses only sites as centers.

Algorithm 1 Greedy $k$-Center

1.0 procedure GreedyKCenter1.0 ($S$, $k$, $C^*$, $r^*$) // $C^*$ covers $S$ with optimal radius $r^*$

$C ← \emptyset$

for all $c ∈ C^*$ do

if $c$ is within $r^*$ of some $s ∈ S$ then

Select some $s$ within $r^*$ of $c$

Add $s$ to $C$; delete $s$ from $S$

Delete all $s'$ within $2r^*$ of $s$ from $S$

return $C$

end procedure
Bootstrapping to a Good Algorithm

Use the covering radius $r^*$ and a cover $C^*$ of size $k$ with $r(C^*) = r^*$ to find a good approximation that uses only sites as centers.

Algorithm 2 Greedy $k$-Center 1.0

procedure $\text{GREEDYKCENTER1.0}(S, k, C^*, r^*)$ // $C^*$ covers $S$ with optimal radius $r^*$
\begin{align*}
C &\leftarrow \emptyset \\
\text{for all } c \in C^* \text{ do} \\
&\quad \text{if } c \text{ is within } r^* \text{ of some } s \in S \text{ then} \\
&\quad\quad \text{Select some } s \text{ within } r^* \text{ of } c \\
&\quad\quad \text{Add } s \text{ to } C; \text{ delete } s \text{ from } S \\
&\quad\quad \text{Delete all } s' \text{ within } 2r^* \text{ of } s \text{ from } S \\
&\quad \text{return } C \\
\end{align*}
end procedure
Better Bootstrapping : Dropping $C^*$

The selected sites form a cover of $S$ of radius $2r^*$
Better Bootstrapping: Dropping $C^*$

The selected sites form a cover of $S$ of radius $2r^*$

In fact, we don’t need $C$!

Algorithm 4

Greedy $k$-Center

1.5

procedure GreedyKCenter1.5 ($S$, $k$, $r^*$) // The optimal radius for $k$ centers is $r^*$

$C \leftarrow \emptyset$

while $S \neq \emptyset$

do

Select some $s \in S$

Add $s$ to $C$; delete $s$ from $S$

Delete all $s'$ within $2r^*$ of $s$ from $S$

return $C$

end procedure

Knowing only $r^*$ can yield a 2-approximation
**Better Bootstrapping : Dropping C**

The selected sites form a cover of $S$ of radius $2r^*$

In fact, we don’t need $C$!

Just pick next $s_i$ to be more than distance $2r^*$ from any site already in $C$. 

---

**Algorithm 5**

```
procedure GreedyKCenter1.5 (S, k, r*)
    // The optimal radius for $k$ centers is $r^*$
    C ← ∅
    while S ≠ ∅
        Select some $s_t$ ∈ S
        Add $s_t$ to $C$; delete $s_t$ from S
        Delete all $s_t'$ within $2r^*$ of $s_t$ from S
    return $C$
end procedure
```

Knowing only $r^*$ can yield a 2-approximation
Better Bootstrapping : Dropping $C^*$

The selected sites form a cover of $S$ of radius $2r^*$
In fact, we don’t need $C$!
Just pick next $s_i$ to be more than distance $2r^*$ from any site already in $C$.

**Algorithm 6** Greedy $k$-Center 1.5

```plaintext
procedure GreedyKCenter1.5($S$, $k$, $r^*$) // The optimal radius for $k$ centers is $r^*$
    $C ← ∅$
    while $S ≠ ∅$ do
        Select some $s ∈ S$
        Add $s$ to $C$; delete $s$ from $S$
        Delete all $s'$ within $2r^*$ of $s$ from $S$
    return $C$
end procedure
```

Knowing only $r^*$ can yield a 2-approximation
**Better Bootstrapping : Dropping $C^*$**

The selected sites form a cover of $S$ of radius $2r^*$
In fact, we don’t need $C$!
Just pick next $s_i$ to be more than distance $2r^*$ from any site already in $C$.

---

**Algorithm 7 Greedy $k$-Center 1.5**

```plaintext
procedure GREEDYKCENTER1.5($S$, $k$, $r^*$) // The optimal radius for $k$ centers is $r^*$
    $C ← ∅$
    while $S ≠ ∅$ do
        Select some $s ∈ S$
        Add $s$ to $C$; delete $s$ from $S$
        Delete all $s'$ within $2r^*$ of $s$ from $S$
    return $C$
end procedure
```

Knowing only $r^*$ can yield a 2-approximation
Even Better Bootstrapping: Dropping \( r^* \)

Idea: Replace "Select some \( s \in S \)" with "Select \( s \in S \) furthest from \( C \)" (initialize \( C \) to any \( s \in S \))
Even Better Bootstrapping: Dropping $r^*$

Idea: Replace "Select some $s \in S$" with "Select $s \in S$ furthest from $C$" (initialize $C$ to any $s \in S$)

Algorithm 9 Greedy $k$-Center 2.0

\begin{verbatim}
procedure GREEDYKCENTER2.0($S, k$)
    if $k \geq |S|$ then
        return $S$
    $C \leftarrow$ some element $s \in S$; delete $s$ from $S$
    while $|C| < k$ do
        Select some $s \in S$ of maximum distance from $C$
        Add $s$ to $C$; delete $s$ from $S$
    return $C$ // Claim: $C$ is a $2r^*$ cover of $S$
end procedure
\end{verbatim}
Why 2.0 Works

**Theorem**
If there is a set \( C' \) of \( k \) centers with covering radius \( r \), then GreedyKCenter2.0 yields a covering \( C \) of \( S \) of size \( k \) with \( r(C) \leq 2r \). Otherwise there is no set \( C' \) of \( k \) centers with \( r(C') = r \).
Why 2.0 Works

Theorem
If there is a set $C'$ of $k$ centers with covering radius $r$, then GreedyKCenter2.0 yields a covering $C$ of $S$ of size $k$ with $r(C) \leq 2r$. Otherwise there is no set $C'$ of $k$ centers with $r(C') = r$.

Proof: It suffices to establish the following (obvious) property
Why 2.0 Works

**Theorem**

If there is a set $C'$ of $k$ centers with covering radius $r$, then GreedyKCenter2.0 yields a covering $C$ of $S$ of size $k$ with $r(C) \leq 2r$. Otherwise there is no set $C'$ of $k$ centers with $r(C') = r$.

**Proof:** It suffices to establish the following (obvious) property

**Property**

If there is a set $C'$ of $k$ centers with $r(C') = r$, then at the end of any iteration of the while loop, either $C$ is a cover of $S$ of radius $2r$ or there is an $s \in S$ of distance greater than $2r$ from $C$. 
Why 2.0 Works

Theorem
If there is a set $C'$ of $k$ centers with covering radius $r$, then GreedyKCenter2.0 yields a covering $C$ of $S$ of size $k$ with $r(C) \leq 2r$. Otherwise there is no set $C'$ of $k$ centers with $r(C') = r$.

Proof: It suffices to establish the following (obvious) property

Property
If there is a set $C'$ of $k$ centers with $r(C') = r$, then at the end of any iteration of the while loop, either $C$ is a cover of $S$ of radius $2r$ or there is an $s \in S$ of distance greater than $2r$ from $C$.

Thus, after $k$ iterations, either $C$ is a cover of radius $2r$, or $C \cup \{s\}$ is a set of $k + 1$ elements, all of which are pairwise more than distance $2r$ apart, contradicting existence of $C'$
Why 2.0 Works

Theorem
If there is a set $C'$ of $k$ centers with covering radius $r$, then GreedyKCenter2.0 yields a covering $C$ of $S$ of size $k$ with $r(C) \leq 2r$. Otherwise there is no set $C'$ of $k$ centers with $r(C') = r$.

Proof: It suffices to establish the following (obvious) property

Property
If there is a set $C'$ of $k$ centers with $r(C') = r$, then at the end of any iteration of the while loop, either $C$ is a cover of $S$ of radius $2r$ or there is an $s \in S$ of distance greater than $2r$ from $C$.

Thus, after $k$ iterations, either $C$ is a cover of radius $2r$, or $C \cup \{s\}$ is a set of $k + 1$ elements, all of which are pairwise more than distance $2r$ apart, contradicting existence of $C'$

Corollary
GreedyKCenter2.0 produces a set $C$ of $k$ centers with $r(C) \leq 2r^*$