Coresets

Mariette Yvinec

MPRI 2009-2010, C2-14-1, Lecture 3b
ENSL Winter School, January 2010
Definition of Coresets

Example: Coresets for the MEB

Minimum enclosing ball

Let $\mathcal{P}$ a set of $n$ points in $\mathbb{R}^d$. The minimum enclosing ball of $\mathcal{P}$, $\text{MEB}(\mathcal{P})$ is the ball with minimum radius whose closure contains all the points in $\mathcal{P}$.

Complexity

Finding the MEB of a set of $n$ points in $\mathbb{R}^d$ is an LP-type problem: it can be solved in $O(n)$ but there is no algorithm with complexity polynomial wrt $d$. 
Definition of Coresets

Example: Coresets for the MEB

Coreset for MEB

$\mathcal{P}$ a set of $n$ points in $\mathbb{R}^d$, $r(\mathcal{P})$ the radius of MEB($\mathcal{P}$)

There exist a subset $\mathcal{P}' \subset \mathcal{P}$ st:
- the size of $\mathcal{P}'$ is less than $\frac{2}{\epsilon}$
- the center $c(\mathcal{P}')$ of MEB($\mathcal{P}'$) satisfies $d(p, c(\mathcal{P}')) \leq (1 + \epsilon)r(\mathcal{P})$, $\forall p \in \mathcal{P}$

Such a subset is a coreset of $\mathcal{P}$ for MEB.

More generally

For a set $\mathcal{P}$ of $n$ points in $\mathbb{R}^d$ and a given problem.
A coreset is a subset $\mathcal{P}'$ of $\mathcal{P}$ such that:
- the size of $\mathcal{P}'$ does not depend on $d$ or $n$
- the solution for $\mathcal{P}'$ is an approximation of the solution for $\mathcal{P}$.

$\epsilon$-coreset: the solution for $\mathcal{P}'$ is within $\epsilon$ of the solution for $\mathcal{P}$
An optimization problem

\( f(x) \) is a concave function on \( \mathbb{R}^n \),

\[
    f(\alpha x + (1 - \alpha)y) \geq \alpha f(x) + (1 - \alpha)f(y)
\]

\( \tau_u \) is the unity simplex: \( \{ x \in \mathbb{R}^n : x_i \geq 0, \sum x_i = 1 \} \)

\[
    \max_x f(x)
\]

subject to \( x \in \tau_u \)

A greedy algorithm provides sparse approximations of the optimum and coresets for various problems such as smallest distance to a polytope, MEB, SVM training
Algorithm 1.

1. Start with $x(0) := \text{argmax } f[e_i]$ for $e_i$ vertex of $\tau_u$.
2. For $k = 0, \ldots, \kappa$ find $x(k + 1)$ from $x(k)$ as follows
   - $i' := \text{argmax}_i \{ e_i^\top \nabla f(x(k)) \}$
   - $\alpha' := \text{argmax}_{\alpha \in [0, 1]} f [x(k) + \alpha (e_{i'} - x(k))]$
   - $x(k + 1) := x(k) + \alpha'(e_{i'} - x(k))$

Frank-Wolfe algorithm

Maximizes concave $f$ on polytope $F$.
At each step
1. find $y' = \text{argmax}_{y \in F} f(x(k)) + (y - x(k))^\top \nabla f(x(k))$
2. find $x(k + 1)$ as the optimal $x \in [x(k), y']$

Algorithm 1. is a particular case of Frank-Wolfe algorithm:
when $F = \tau_u$, $y' = e_{i'}$ if $i' = \text{argmax}_i \{ e_i^\top \nabla f(x(k)) \}$
The Wolfe dual

Primal

\[ \max_{x \in \mathbb{R}^n} f(x) \]
subject to \( x \in \tau_u \)

Dual

\[ \min_{z \in \mathbb{R}, x \in \mathbb{R}^n} z + f(x) - x^\top \nabla f(x) \]
subject to \( z \geq \max_i e_i^\top \nabla f(x) \)

\[ \min_{x \in \mathbb{R}^n} w(x) \]
\[ w(x) = z(x) + f(x) - x^\top \nabla f(x) \]
\[ z(x) = \max_i e_i^\top \nabla f(x) \]
The Wolfe dual

\[ \min_{x \in \mathbb{R}^n} w(x) = z(x) + f(x) - x^T \nabla f(x) \]

\[ z(x) = \max_i e_i^T \nabla f(x) \]

\[ w(x) = f(x) + (e_{i'} - x)^T \nabla f(x) \]

with \( i' = \arg \max_i \{ e_i^T \nabla f(x) \} \)

\[ w(x) = \max_{y \in \tau} f(x) + (y - x) \nabla f(x) \]

\[ w(x) = \max_{y \in \tau} I f_x(y) \]

\( I f_x \) the linear approximation of \( f \) at point \( x \)

If \( x^* \) is the optimal point of primal, \( x^{**} \) the optimal point of dual

In fact, strong duality holds:

\[ w(x) \geq w(x^{**}) \geq f(x^*) \geq f(x) \]

\[ w(x) \geq w(x^*) = f(x^*) \geq f(x) \]
The constant $C_f$

$f$ is assumed to be continuously differentiable.

$C_f$ measures the non linearity of $f$

$C_f$ is related to the Bregman distance defined by $f$

\[
C_f = \sup \frac{1}{\alpha^2} \left[ f(x) + (y - x)^\top \nabla f(x) - f(y) \right]
\]

sup taken over $x, z, \alpha$ with $y = x + \alpha (z - x) \in S$

Taylor expansion yields

\[
f(x + \alpha(z - x)) = f(x) + \alpha (z - x)^\top \nabla f(x) + \frac{1}{2} \alpha^2 (z - x)^\top \nabla^2 f(\bar{x})(z - x)
\]

\[
C_f \leq \sup_{x, z \in \tau_u, \bar{x} \in [x, z]} -\frac{1}{2} (z - x)^\top \nabla^2 f(\bar{x})(z - x)
\]
The primal/dual approximation theorems

\[
\text{primal error: } h(x) = \frac{1}{4C_f} \left[ f(x^*) - f(x) \right],
\]
\[
\text{gap: } g(x) = \frac{1}{4C_f} \left[ w(x) - f(x) \right]
\]

Primal/dual theorems

If function \( f \) is continuously differentiable

**Theorem 1** At each iteration of Algorithm 1, \( h(x(k + 1)) \leq h(x(k)) - g(x(k))^2 \).

**Theorem 2** Iterate \( x(k) \in k\)-face of \( \tau_u \) and \( h(x(k)) \leq \frac{1}{k+3} \).

**Theorem 3** Let \( \epsilon > 0 \) and \( \kappa = \left\lceil \frac{1}{\epsilon} \right\rceil \)
\[
\exists \hat{k} \in [\kappa, 2\kappa], \text{ such that } g(x(\hat{k})) \leq \epsilon.
\]
Proof of primal/dual approximation Th1

Th1: At each iteration, $h(x(k+1)) \leq h(x(k)) - g(x(k))^2$.

Let $x \in \tau_u$, $i' := \arg\max_i \{e_i^T \nabla f(x)\}$

$w(x) = \max_{z \in \tau_u} l f_x(z) = f(x) + (e_{i'} - x)^T \nabla f(x)$.

Let $y = x + \alpha (e_{i'} - x)$ with $\alpha \in [0, 1]$.

\[
\begin{align*}
    f(y) & \geq f(x) + (y - x)^T \nabla f(x) - \alpha^2 C_f, \text{ (by definition of } C_f) \\
    & \geq f(x) + \alpha (e_{i'} - x)^T \nabla f(x) - \alpha^2 C_f, \\
    & \geq f(x) + \alpha (w(x) - f(x)) - \alpha^2 C_f.
\end{align*}
\]

\[
\begin{align*}
h(y) & = \frac{1}{4C_f} [f(x^*) - f(y)] \\
    & \leq h(x) - \frac{\alpha}{4C_f} (w(x) - f(x)) + \frac{\alpha^2}{4} \\
    & \leq h(x) - \alpha g(x) + \frac{\alpha^2}{4},
\end{align*}
\]
Proof of primal/dual approximation Th1

Th1: At each iteration, \( h(x(k + 1)) \leq h(x(k)) - g(x(k))^2 \).

\[
\begin{align*}
\forall x \in \tau_u \text{ and } \alpha \in [0, 1] \\
\text{if } i' := \arg \max_i \{e_i^T \nabla f(x)\} \\
\text{and } y = x + \alpha (e_{i'} - x) \\
\end{align*}
\]

\[
h(y) \leq h(x) - \alpha g(x) + \frac{\alpha^2}{4} \quad (1)
\]

If \( x = x(k) \) and \( \alpha = \arg \max \{f(x + \alpha (e_{i'} - x))\} \), \( y = x(k + 1) \).
Then \( \forall \alpha \in [0, 1], \ h(x(k + 1)) \leq h(x(k)) - \alpha g(x(k)) + \frac{\alpha^2}{4} \).
Th1 then follows from the choice \( \alpha = 2g(x(k)) \) possible if \( g(x(k)) \leq \frac{1}{2} \).

\( g(x(k)) \leq \frac{1}{2} \) results from the choice of \( x(0) \):
If \( g(x(k)) \geq \frac{1}{4}, \ h(x(k)) + \alpha (e_{i'} - x(k)) \leq h(x(k)), \ \forall \alpha \in [0, 1]. \)
In particular, \( h(e_{i'}) \leq h(x(k)) \Leftrightarrow f(e_{i'}) \geq f(x(k)) \),
which contradicts: \( f(x(0)) \geq f(e(i')) \) and \( f(x(k)) \) increasing with \( k \).
Proof of primal/dual approximation Th2

Theorem 2: Iterate $x(k) \in k$-face of $\tau_u$ and $h(x(k)) \leq \frac{1}{k+3}$.

- $x(k)$ is combination of at most $k + 1$ vertices of $\tau_u$.
- From Th1 and $\forall x$, $h(x) \leq g(x)$

\[
h(x(k+1)) \leq h(x(k)) - h(x(k))^2 \\
\leq h(x(k))(1 - h(x(k))) \leq \frac{h(x(k))}{1 + h(x(k))}
\]

Then Th2 follows by induction. \hfill \square
Proof of primal/dual approximation Th3

Th3: Let $\epsilon > 0$ and $\kappa = \left\lfloor \frac{1}{\epsilon} \right\rfloor$, $\exists \hat{k} \in [\kappa, 2\kappa]$, such that $g(x(\hat{k})) \leq \epsilon$.

From th2, $\forall k \geq \kappa$, $h(x(k)) \leq \epsilon$.

Then from th1, $h(x(k + 1)) \leq h(x(k)) - g(x(k))^2$

thus either $g(x(k)) \leq \epsilon$ or $h(x(k + 1)) \leq h(x(k)) - \epsilon^2$.

If only the second case happens, $h(x(2\kappa))$ becomes negative. □
Sparse approximation and coresets

Coresets for the optimization problem

An $\epsilon$-coreset for the problem $\max_{x \in \tau_u(\mathbb{R}^n)} f(x)$ is a subset $N \subset [1, \ldots, n]$ of coordinates, such that the optimal point $x^*(N) = \arg\max_{x \in \tau_u(\mathbb{R}^N)} f(x)$ satisfies $w(x^*(N)) - f(x^*(N)) \leq 4\epsilon C_f$.

Sparse approximation

In $O(\frac{1}{\epsilon})$ iterations, Algorithm 1. provides a point $x'$ such that $w(x') - f(x') \leq 4\epsilon C_f$ with a small subset $N' \subset [1, \ldots, n]$ of non null coordinates. But $N'$ is not a coreset because the restricted dual $w_{N'}(x) \neq w(x)$. Therefore we can have that $w(x^*(N')) \gg w(x')$.

To get an $\epsilon$-coreset:
- either run Algorithm 1, for $O(\frac{1}{\epsilon^2})$ iterations
- or run Algorithm 2, $O(\frac{1}{\epsilon})$ iterations.
Theorem
If function $f$ is continuously differentiable, after $\kappa = O\left(\frac{1}{\epsilon^2}\right)$ iterations, Algorithm 1 provides an approximate solution $x(\kappa)$ whose subset $N$ of non null coordinates is an $\epsilon$-coreset.

From Th1, $\forall x \in \tau, g(x) \leq \sqrt{h(x)}$
by def., $f(x^*(N)) \geq f(x(\kappa)) \iff h(x^*(N)) \leq h(x(\kappa))$ 
From Th2, $h(x(\kappa)) \leq \frac{1}{\kappa+3} \leq \frac{1}{\epsilon^2}$
\[\Rightarrow g(x^*(N)) \leq \frac{1}{\epsilon}\]
Getting Coresets

Algorithm 2.

1. Start with $i' := \text{argmax}_i f(e_i)$, $N(0) = \{i'\}$.

2. For $k = 0, \ldots, \kappa$ find $N(k+1)$ from $N(k)$ as follows
   - If $g(x^*(N(k))) \leq \epsilon$ return $N(k)$.
   - $i' := \text{argmax}_i e_i^\top \nabla f(x^*(N(k)))$
   - $N(k+1) := N(k) \cup \{i'\}$

Theorem
Algorithm 2. yields an $\epsilon$-coreset after $\kappa = \frac{2}{\epsilon}$ iterations.

Proof
Let $x = x(N(k))$ and $i' := \text{argmax}_i e_i^\top \nabla f(x)$. Then $h(x(N(k+1))) \leq h(x + \alpha(e_{i'} - x) \leq h(x) - g(x)^2$, $\forall \alpha \in [0, 1]$. This is Th1 for $x^*(N(k))$. Th2 and Th3 apply to $x^*(N(k))$. Therefore $\exists k \in [\kappa/2, \kappa]$ such that $g(x^*(N(k))) \leq \epsilon$.

□
Polytope distance

Distance from a point \( o \) to a polytope \( \text{conv}(\mathcal{P}) \)

\[
\mathcal{P} \in \mathbb{R}^d = \{p_1, \ldots, p_n\}, \quad \mathcal{P} = [p_1, \ldots, p_n],
\]
\[
p \in \text{conv}(\mathcal{P}) = \sum_i x_i p_i = P x \quad \leftarrow \quad x \in \tau_u \text{ of } \mathbb{R}^n
\]

\[
d(o, \text{conv}(\mathcal{P}))^2 = \min_{p \in \text{conv}(\mathcal{P})} p^T p = \min_{x \in \tau_u} x^T P^T P x
\]

\[
f(x) = -x^T P^T P x
\]

\[
\nabla f(x) = -2 P^T P x
\]

a point in \( \mathcal{P} \) \( \leftarrow \) a vertex of \( \tau_u \)

a subset of \( \mathcal{P} \) \( \leftarrow \) a face of \( \tau_u \)

\[
\min_i p_i^T p \quad \leftarrow \quad \max_i e_i^T \nabla f(x)
\]

Algorithm 1 = Algorithme de Gilbert.
Polytope distance

\[ f(x) = -x^\top P^\top Px \]
\[ \nabla f(x) = -2P^\top Px \]

\[ C_f = \sup_{x,y \in \tau} (x - y)^\top P^\top P(x - y) \]
\[ C_f = \sup_{p,q \in P} \|p - q\|^2 = \text{diam}(P)^2 \]

\[ D = \text{diam}(P), \quad \delta = d(o, \text{conv}(P)), \]
\[ \frac{1}{\epsilon} \text{ iterations for an approximation of } \delta^2 \text{ within } 4D^2\epsilon: \]

\[ \|p\|^2 - \|p^*\|^2 \leq 4D^2\epsilon \implies \|p\| - \|p^*\| \leq 2 \frac{D^2}{\|p^*\|} \frac{\epsilon}{\delta^2} \]
\[ \|p\| \leq \left( 1 + 2\epsilon \frac{D^2}{\delta^2} \right) \|p^*\| \]
**Polytope distance (bis)**

\[
f(x) = -\|Px\| = -\sqrt{x^\top P^\top Px}
\]

\[
\nabla f(x) = -\frac{P^\top Px}{\|Px\|}
\]

\[
w(x) = \max_i e_i^\top \nabla f(x)
\]

\[
w(x) = \min_i \frac{p_i^\top p}{\|p\|}
\]

\[
\nabla^2 f = \frac{P^\top P}{\|Px\|} - \frac{P^\top Pxx^\top P^\top P}{\|Px\|^3}
\]

\[
C_f \leq \sup_{x, y \in \tau} \frac{(x - y)^\top P^\top P(x - y)}{\delta} \leq \frac{D^2}{\delta}
\]

After \(\frac{1}{\epsilon}\) iterations

\[
\|p\| - \|p^*\| \leq 4 \frac{D^2}{\delta} \epsilon \implies \|p\| \leq \left(1 + 4\epsilon \frac{D^2}{\delta^2}\right) \|p^*\|
\]
Minimum enclosing ball

\[ \mathcal{P} \in \mathbb{R}^d = \{p_1, \ldots, p_n\} \]

\[ P = [p_1, \ldots, p_n] \]

\[ b^\top = [p_1^2, \ldots, p_n^2] \]

\[ \text{conv}(\mathcal{P}) \iff \tau_u \text{ of } \mathbb{R}^n \]

\[ p = \sum_i x_i p_i = Px \iff x \in \tau_u \]

**Primal:** \( \max_{x \in \tau_u} f(x), \quad f(x) = b^\top x - x^\top P^\top Px \)

\( \nabla f(x) = b^\top - 2P^\top Px, \quad f(x) - x^\top \nabla f(x) = x^\top P^\top Px \)

\( e_i^\top \nabla f(x) = p_i^2 - 2p_i^\top Px \)

**Dual problem:** \( \min_{x \in \tau_u} w(x), \quad w(x) = \max_i (p_i^2 - 2p_i^\top Px) + x^\top P^\top Px \)

\[ \iff \min_{p = Px \in \text{conv}(\mathcal{P})} \max_i (p_i^2 - 2p_i p + p^2) = \max_i (p_i - p)^2 \]

\( p^* = Px^* \) is the center of MEB, \( w(x^*) \) is the square radius of MEB
Minimum enclosing ball

\[ f(x) = b^\top x - x^\top P^\top Px = \sum_i x_ip_i^2 - p^2 \]

\[ \nabla f(x) = b^\top - 2P^\top Px \]

\[ e_i^\top . \nabla f(x) = p_i^2 - 2p_i^\top Px = (p_i - p)^2 - p^2 \]

\[ C_f = \sup_{p,q \in \mathcal{P}} \|p - q\|^2 = \text{diam}(\mathcal{P})^2 = D^2 \]

**Algorithm 1**: Each iteration finds the point \( p_i \) farthest from the current approximation \( p(k) \) and look for the best center in \([p(k), p_i]\)

\[ \frac{2}{\epsilon} \text{ iterations to get an approximate center } p(k) \text{ with } \max_i(p_i - p(k))^2 - r^* \leq 4\epsilon D^2 \text{ or } \max_i(p_i - p(k)) \leq (1 + 2\frac{D^2}{r^* \epsilon})r^* \leq (1 + 8\epsilon)r^*. \]

**Algorithm 2**: Each iteration finds the point \( p_i \) farthest from the current center \( c(k) \) of MEB(\( \mathcal{P}(k) \)) and set \( \mathcal{P}(k+1) = \mathcal{P}(k) \cup \{p_i\} \)

\[ \frac{2}{\epsilon} \text{ iterations to get a subset } \mathcal{P}(k) \text{ whose MEB has center } c(k) \text{ such that } \max_i(p_i - c(k)) \leq (1 + 8\epsilon)r^*. \]
Support vector machine

A classical machine learning problem: classify data points in two classes.

- A training set $\mathcal{P}$ of classified points is given.
- Find a hyperplan separating red and blue points.
- Each data will be classified using this hyperplan.

The best separating hyperplan is the hyperplan with largest margin: largest distance to the nearest training point.

$\textbf{Pb}$ Find the maximal width empty strip between red and blue points.

If general position: $d + 1$ points on the boundary of maximal strip

those points are called support vectors.
Two sets $P$ and $Q$,

Minkovsky sum \[ P \oplus Q = \{ p + q : p \in P, q \in Q \} \]

Minkovsky difference \[ P \ominus Q = \{ p - q : p \in P, q \in Q \} \]

- The Minkovsky sum (difference) of two polytopes is a polytope.
  \[ P = \text{conv}(\mathcal{P}), \; Q = \text{conv}(\mathcal{Q}), \; P \oplus Q = \text{conv} \left( \{ p + q : p \in \mathcal{P}, \; q \in \mathcal{Q} \} \right) \]
  \[ P \ominus Q = \text{conv} \left( \{ p - q : p \in \mathcal{P}, \; q \in \mathcal{Q} \} \right) \]

- $P \ominus Q$ is the set of transalations $t$ s.t. $t + Q \cap P \neq \emptyset$.
  Hence, $o \in P \ominus Q$ iff $Q \cap P \neq \emptyset$. 

Minkovksy Sum
Let $n$ be the unit normal vector to hyperplane $h$. The width of the largest empty strip formed by hyperplanes normal to $n$ is:

$$\min_{p \in P, q \in Q} n^T (p - q)$$

Training SVM problem is:

$$\max_{\|n\| = 1} \min_{p \in P, q \in Q} n^T (p - q) = \max_{t} \min_{p \in P, q \in Q} \frac{(p - q)^T t}{\|t\|}$$

which is just the Wolfe dual of:

$$\min_{t \in \mathcal{O}} \|t\| = \min_{t \in \mathcal{O}} \sqrt{t^T t}.$$