Submodular Optimization: From Discrete to Continuous and Back

Hamed Hassani                              Amin Karbasi

http://iid.yale.edu/icml/icml-20.md/
We consider the following optimization problem:

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]
Non-monotone and DR-submodular

measured continuous greedy:

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:
  \[
  v_t = \arg \max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle \\
  \text{subject to } v \leq u - x_t
  \]
  \[
  x_{t+1} = x_t + \frac{1}{T} v_t
  \]


Non-monotone and DR-submodular

- measured continuous greedy:
  - Initialize at $x_0 = 0$
  - Repeat for $T$ iterations:
    $$v_t = \arg \max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle$$
    $$v \leq u - x_t$$
    $$x_{t+1} = x_t + \frac{1}{T}v_t$$


Non-monotone and DR-submodular

measured continuous greedy:

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:

  $$v_t = \arg\max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle$$

  $$v \leq u - x_t$$

  $$x_{t+1} = x_t + \frac{1}{T} v_t$$


Non-monotone and DR-submodular

- measured continuous greedy:
  - Initialize at $x_0 = 0$
  - Repeat for $T$ iterations:
    $$v_t = \arg \max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle$$
    subject to $v \leq u - x_t$
    $$x_{t+1} = x_t + \frac{1}{T}v_t$$


Non-monotone and DR-submodular

measured continuous greedy:

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:

$$v_t = \arg \max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle \quad \text{s.t.} \quad v \leq u - x_t$$

$$x_{t+1} = x_t + \frac{1}{T}v_t$$

[Bian, Levy, Krause, Buhmann]

The measured continuous greedy algorithm achieves $\frac{1}{\epsilon} \frac{1}{\epsilon} \text{OPT}$ for constrained maximization of non-monotone DR-submodular functions

“Continuous DR-submodular Maximization: Structure and Algorithms”, NIPS’17
Non-monotone and DR-submodular

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:

  $$ v_t = \arg \max_{v \in \mathcal{K}, v \leq 1 - x_t} \langle \nabla F(x_t), v \rangle $$

  $$ x_{t+1} = x_t + \frac{1}{T} v_t $$

  $$ d_{t+1} = (1 - \rho_t) d_t + \rho_t g_t $$
Non-monotone and DR-submodular

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:
  
  $v_t = \arg\max_{v \in K} \langle \nabla F(x_t), v \rangle$
  
  $v \leq 1 - x_t$
  
  $x_{t+1} = x_t + \frac{1}{T} v_t$
  
  $d_{t+1} = (1 - \rho_t) d_t + \rho_t g_t$
Non-monotone and DR-submodular

- Initialize at $x_0 = 0$
- Repeat for $T$ iterations:
  \[ v_t = \arg \max_{v \in \mathcal{K}} \langle \nabla F(x_t), v \rangle \quad v \leq 1 - x_t \]
  \[ x_{t+1} = x_t + \frac{1}{T} v_t \]
  \[ d_{t+1} = (1 - \rho_t) d_t + \rho_t g_t \]

The stochastic measured continuous greedy algorithm achieves with sample complexity $O\left(\frac{1}{\epsilon^2}\right)$

\[ \frac{1}{e} \text{OPT} - \epsilon \]

“Stochastic Conditional Gradient Methods: From Convex Minimization to Submodular Maximization”, JMLR ’20
“Stochastic Conditional Gradient++: (Non-)Convex Minimization and Continuous Submodular Maximization”, preprint ’20

Relevant work:
“Non-monotone DR-submodular Maximization: Approximation and Regret Guarantees”, Dürr, Nguyen Kim Thang, Abhinav Srivastav, Léo Tible, 2019
Non-monotone and DR-submodular

Constrained case:

\[ \frac{1}{e} \quad \text{?} \quad 0.491 \]

Hard
Non-monotone and DR-submodular

Constrained case:

\[ \frac{1}{e} \quad ? \quad 0.491 \quad \text{Hard} \]
Non-monotone and DR-submodular

• Constrained case:

• Unconstrained case:

\[
\frac{1}{e} \quad 0.491
\]

[ Bian, Buhmann, Krause ]

For the case where \( \mathcal{K} \) is a box, an extension of the double-greedy method (covered in part 1) achieves a tight guarantee \( \frac{1}{2} \text{OPT} \).

“Optimal Continuous DR-Submodular Maximization and Applications to Provable Mean Field Inference”, JMLR ‘20
Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]
Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]
Continuous Submodular Maximization

maximize $F(x)$

subject to: $x \in \mathcal{K}$
Continuous Submodular Maximization

maximize $F(x)$

subject to: $x \in K$
Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]

Very recent progress: Coordinate-ascent methods
Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } ||x||_1 \leq B
\]

- Very recent progress: Coordinate-ascent methods
Continuous Submodular Maximization

maximize $F(x)$
subject to: $||x||_1 \leq B$

Basic idea:

while the budget $B$ is not exhausted

for each coordinate $i$:

find: $\alpha_i = \arg \max_{\alpha \geq 0} \frac{F(x + \alpha e_i) - F(x)}{\alpha}$

choose coordinate $i^*$ with the largest $\alpha_{i^*}$

update $x = x + \alpha_{i^*}e_{i^*}$
Continuous Submodular Maximization

maximize $F(x)$

subject to: $||x||_1 \leq B$

Basic idea:

while the budget $B$ is not exhausted

for each coordinate $i$:

find: $\alpha_i = \arg \max_{\alpha \geq 0} \frac{F(x + \alpha e_i) - F(x)}{\alpha}$

choose coordinate $i^*$ with the largest $\alpha_{i^*}$

update $x = x + \alpha_{i^*} e_{i^*}$
Continuous Submodular Maximization

maximize $F(x)$

subject to: $||x||_1 \leq B$

Basic idea:

while the budget $B$ is not exhausted

for each coordinate $i$:

find: $\alpha_i = \arg \max_{\alpha \geq 0} \frac{F(x + \alpha e_i) - F(x)}{\alpha}$

choose coordinate $i^*$ with the largest $\alpha_{i^*}$

update $x = x + \alpha_{i^*} e_{i^*}$

$F(x + \alpha e_i)$
Continuous Submodular Maximization

maximize $F(x)$
subject to: $||x||_1 \leq B$

Basic idea:

while the budget $B$ is not exhausted

for each coordinate $i$:

find: $\alpha_i = \arg \max_{\alpha \geq 0} \frac{F(x + \alpha e_i) - F(x)}{\alpha}$

choose coordinate $i^*$ with the largest $\alpha_i^*$

update $x = x + \alpha_i^* e_i^*$

Coordinate descent methods obtain a $(1 - \frac{1}{e} - \epsilon) \text{OPT}$ solution to the above problem with computational complexity $O\left(\frac{n^3}{\epsilon^{2.5}}\right)$

[Fieldmann, Karbasi]
Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]
Continuous Submodular Maximization

\[
\begin{align*}
\text{maximize} & \quad F(x) \\
\text{subject to:} & \quad x \in \mathcal{K}
\end{align*}
\]
Continuous Submodular Maximization

maximize $F(x)$
subject to: $x \in K$

Many open questions:
General convex constraints, stochastic, better algorithms, etc
Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in \mathcal{K}
\]
Submodular Maximization

maximize $F(x)$
subject to: $x \in \mathcal{K}$
Continuous Submodular Maximization

\begin{align*}
\text{maximize } & F(x) \\
\text{subject to: } & x \in \mathcal{K}
\end{align*}
Continuous Submodular Maximization

maximize $F(x)$

subject to: $\mathbf{x} \in [0, 1]^n$
Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in [0, 1]^n
\]

- Initialize \( x = (0, 0, \ldots, 0) \) \( y = (1, 1, \ldots, 1) \)
- For \( i = 1 \) to \( n \)
  
  Find \( z_i \in [0, 1] \) as the value of the final solution at the \( i \)-th coordinate

Continuous Submodular Maximization

maximize $F(x)$

subject to: $x \in [0, 1]^n$

- Initialize $x = (0, 0, \ldots, 0)$
  $y = (1, 1, \ldots, 1)$

- For $i = 1$ to $n$
  
  Find $z_i \in [0, 1]$ as the value of the final solution at the $i$-th coordinate

Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in [0, 1]^n
\]

- Initialize \( x = (0, 0, \cdots, 0) \) \( y = (1, 1, \cdots, 1) \)

- For \( i = 1 \) to \( n \)

  Find \( z_i \in [0, 1] \) as the value of the final solution at the \( i \)-th coordinate

  \[
  x = (z_1, 0, \cdots, 0) \\
y = (z_1, 1, \cdots, 1)
  \]
Continuous Submodular Maximization

\[
\text{maximize } F(x) \\
\text{subject to: } x \in [0, 1]^n
\]

- Initialize \( x = (0, 0, \cdots, 0) \)
  \( y = (1, 1, \cdots, 1) \)

- For \( i = 1 \) to \( n \)
  
  Find \( z_i \in [0, 1] \) as the value of the final solution at the \( i \)-th coordinate
  
  \( x = (z_1, z_2, 0, \cdots, 0) \)
  \( y = (z_1, z_2, 1, \cdots, 1) \)

Continuous Submodular Maximization

maximize $F(x)$

subject to: $x \in [0, 1]^n$

- Initialize $x = (0, 0, \ldots, 0)$
  $y = (1, 1, \ldots, 1)$

- For $i = 1$ to $n$

  Find $z_i \in [0, 1]$ as the value of the final solution at the $i$-th coordinate

  $x = (z_1, z_2, \ldots, z_{i-1}, z_i, 0, \ldots, 0)$
  $y = (z_1, z_2, \ldots, z_{i-1}, z_i, 1, \ldots, 1)$
Continuous Submodular Maximization

\[
\begin{align*}
\text{maximize } & \quad F(x) \\
\text{subject to: } & \quad x \in [0, 1]^n
\end{align*}
\]

- Initialize \( x = (0, 0, \ldots, 0) \)
  \( y = (1, 1, \ldots, 1) \)
- For \( i = 1 \) to \( n \)
  
  Find \( z_i \in [0, 1] \) as the value of the final solution at the \( i \)-th coordinate

\[
\begin{align*}
x &= (z_1, z_2, \ldots, z_{i-1}, z_i, 0, \ldots, 0) \\
y &= (z_1, z_2, \ldots, z_{i-1}, z_i, 1, \ldots, 1)
\end{align*}
\]

\( z_i \) is found by drawing an analogy to a two-player zero-sum game

Continuous Submodular Maximization

maximize $F(x)$
subject to: $x \in [0, 1]^n$

Initialize
\[ x = (0, 0, \cdots , 0) \]
\[ y = (1, 1, \cdots , 1) \]

For $i = 1$ to $n$

Find $z_i \in [0, 1]$ as the value of the final solution at the $i$-th coordinate

\[ x = (z_1, z_2, \cdots , z_{i-1}, z_i, 0, \cdots , 0) \]
\[ y = (z_1, z_2, \cdots , z_{i-1}, z_i, 1, \cdots , 1) \]

$z_i$ is found by drawing an analogy to a two-player zero-sum game

Output: $(z_1, z_2, \cdots , z_n)$
Continuous Submodular Maximization

maximize $F(x)$
subject to: $x \in \mathcal{K}$

For the case where $\mathcal{K}$ is a box, the above algorithm achieves a tight guarantee $\frac{1}{2} \text{OPT} - \epsilon$ with complexity $O\left(\frac{n^2}{\epsilon}\right)$.

“Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization”, NIPS ‘18
Continuous Submodular Maximization

maximize $F(x)$
subject to: $x \in \mathcal{K}$

[Niazadeh, Roughgarden, Wang]

For the case where $\mathcal{K}$ is a box, the above algorithm achieves a tight guarantee $\frac{1}{2} \text{OPT} - \epsilon$ with complexity $O\left(\frac{n^2}{\epsilon}\right)$.

“Optimal Algorithms for Continuous Non-monotone Submodular and DR-Submodular Maximization”, NIPS ‘18

Many open questions:
non-monotone and constrained, stochastic, etc
Submodular Maximization

maximize $F(x)$
subject to: $x \in K$
Submodular Maximization

maximize $F(x)$
subject to: $x \in \mathcal{K}$

Oracle
- Perfect
- Imperfect
  - Stochastic
  - Online
  - Bandit

Structure
- DR-submodular
- Submodular
Bridging Discrete and Continuous Settings

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{bridges} \quad \max_{x \in \mathcal{K}} F(x) \]

“Maximizing a monotone submodular function subject to a matroid constraint”,
Calinescu, Chekuri, Pal, Vondrák, 2011
Bridging Discrete and Continuous Settings

Maximizing a monotone submodular function subject to a matroid constraint, Calinescu, Chekuri, Pal, Vondrák, 2011
Bridging Discrete and Continuous Settings

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{submodular} \quad \text{maximizing a monotone submodular function subject to a matroid constraint}, \]

Calinescu, Chekuri, Pal, Vondrák, 2011

\[ \max_{x \in \mathcal{K}} F(x) \quad \text{DR-submodular} \]
Bridging Discrete and Continuous Settings

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{submodular} \]

\[ \max_{x \in \mathcal{K}} F(x) \quad \text{DR-submodular} \]

cardinality
matroid
knapsack

“Maximizing a monotone submodular function subject to a matroid constraint”, Calinescu, Chekuri, Pal, Vondrák, 2011
Bridging Discrete and Continuous Settings

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{submodular} \]

\[ \max_{x \in \mathcal{K}} F(x) \quad \text{DR-submodular} \]

Cardinality
Matroid
Knapsack

Polytope

“Maximizing a monotone submodular function subject to a matroid constraint”, Calinescu, Chekuri, Pal, Vondrák, 2011
Bridging Discrete and Continuous Settings

\[
\max_{S \in \mathcal{I}} f(S)
\]

 cardinality
 matroid
 knapsack

\[
\max_{x \in \mathcal{K}} F(x)
\]

DR-submodular
 polytope

"Maximizing a monotone submodular function subject to a matroid constraint", Calinescu, Chekuri, Pal, Vondrák, 2011
“Maximizing a monotone submodular function subject to a matroid constraint”,
Calinescu, Chekuri, Pal, Vondrák, 2011
Bridging Discrete and Continuous Settings

Building blocks:

- Multi-linear extension
- Equivalent formulations
- Rounding

"Maximizing a monotone submodular function subject to a matroid constraint",
Calinescu, Chekuri, Pal, Vondrák, 2011
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \quad \rightarrow \quad F : \mathcal{X} \rightarrow \mathbb{R} \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \quad \Rightarrow \quad F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \chi \rightarrow \mathbb{R} \]

\[ \chi = [0, 1]^n \]
**Multi-linear Extension**

\[ f : 2^V \to \mathbb{R} \]

\[ V = \{1, 2, \ldots, n\} \]

\[ F : \mathcal{X} \to \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ 2^V = \{0, 1\}^n = \text{corner points of } [0, 1]^n \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ 2^V = \{0, 1\}^n = \text{corner points of } [0, 1]^n \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ \mathbf{x} \in [0, 1]^n \]
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]
\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \to \mathbb{R} \]
\[ \mathcal{X} = [0, 1]^n \]

\[ x \in [0, 1]^n \]
\[ x = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]
\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]
\[ \mathcal{X} = [0, 1]^n \]
\[ \mathbf{x} \in [0, 1]^n \]
\[ \mathbf{x} = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[ F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]
\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \to \mathbb{R} \]
\[ \mathcal{X} = [0, 1]^n \]
\[ \mathbf{x} \in [0, 1]^n \]
\[ \mathbf{x} = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]
\[ F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]
\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)] \]
Mul/-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ x \in [0, 1]^n \]

\[ x = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[ F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]

\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)] \]

\[ 1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n \]
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \to \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ \mathbf{x} \in [0, 1]^n \]

\[ \mathbf{x} = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[
F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)
\]

\[
= \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)]
\]

random set \( S = \{ \} \)
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]
\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]
\[ \mathcal{X} = [0, 1]^n \]

\[ \mathbf{x} \in [0, 1]^n \]
\[ \mathbf{x} = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[ F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]
\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)] \]

random set \( S = \{1, 2, 3, \cdots, i, \cdots, n\} \)
\[ \text{wp} x_1 \]
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \to \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ x \in [0, 1]^n \]

\[ x = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[ F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]

\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)] \]

random set \( S = \{ \}

1 \quad 2 \quad 3 \quad \cdots \quad i \quad \cdots \quad n

\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow

wp x_1 \quad x_2 \quad x_3

\}

Yale
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ F : \mathcal{X} \rightarrow \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ x \in [0, 1]^n \]

\[ x = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[
F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i)
\]

\[
= \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)]
\]

random set \( S = \{ \)

1 \( \quad \)
wp \( x_1 \)
2 \( \quad \)
(\( \downarrow \))
3 \( \quad \)
(\( \downarrow \))
\( \cdots \)
\( \cdots \)
\( i \)
\( \cdots \)
\( n \)

(\( \downarrow \))

\( x_i \)

(\( \downarrow \))

\{ \)
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]
\[ V = \{1, 2, \cdots, n\} \]
\[ F : \mathcal{X} \to \mathbb{R} \]
\[ \mathcal{X} = [0, 1]^n \]
\[ x \in [0, 1]^n \]
\[ x = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]
\[ F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]
\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)} [f(S)] \]

random set \( S = \{ \)

\[ \text{wp } x_1 \]
\[ x_2 \]
\[ x_3 \]
\[ \cdots \]
\[ x_i \]
\[ \cdots \]
\[ x_n \]
\[ \} \]
Multi-linear Extension

\[ f : 2^V \rightarrow \mathbb{R} \]

\[ V = \{1, 2, \cdots, n\} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ \mathbf{x} \in [0, 1]^n \]

\[ \mathbf{x} = (x_1, x_2, \cdots, x_n) \quad x_i \in [0, 1] \]

\[ F(\mathbf{x}) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]

\[ = \mathbb{E}_{S \sim (x_1, \cdots, x_m)}[f(S)] \]

random set \( S = \{w, p\} \)

probability of \( S = \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \)
Multi-linear Extension

\[ f : 2^V \to \mathbb{R} \]

\[ V = \{1, 2, \ldots, n\} \]

\[ f(S) \]

\[ F : \mathcal{X} \to \mathbb{R} \]

\[ \mathcal{X} = [0, 1]^n \]

\[ x \in [0, 1]^n \]

\[ x = (x_1, x_2, \ldots, x_n) \quad x_i \in [0, 1] \]

\[ F(x) = \sum_{S \subseteq V} f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \]

\[ = \mathbb{E}_{S \sim (x_1, \ldots, x_m)} [f(S)] \]

random set \( S = \{ \}

probability of \( S = \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \)
Properties of Multi-linear Extension

\[ f \text{ is submodular} \quad \iff \quad F \text{ is DR-submodular} \]
Properties of Multi-linear Extension

\[ f \text{ is submodular} \iff F \text{ is DR-submodular} \]

\[ \nabla^2 F(x) = \begin{pmatrix}
0 & \leq 0 & \leq 0 \\
\leq 0 & 0 & \leq 0 \\
\leq 0 & \leq 0 & 0
\end{pmatrix} \]
Properties of Multi-linear Extension

\[ f \text{ is submodular} \iff F \text{ is DR-submodular} \]

\[ f \text{ is monotone} \iff F \text{ is monotone} \]
Properties of Multi-linear Extension

- $f$ is submodular $\iff F$ is DR-submodular
- $f$ is monotone $\iff F$ is monotone

Multi-linear extension is a linear operator:

$$\alpha f_1 + \beta f_2 \rightarrow \alpha F_1 + \beta F_2$$
Properties of Multi-linear Extension

\[ f \text{ is submodular} \iff F \text{ is DR-submodular} \]

\[ f \text{ is monotone} \iff F \text{ is monotone} \]

Multi-linear extension is a linear operator:
\[ \alpha f_1 + \beta f_2 \rightarrow \alpha F_1 + \beta F_2 \]

\[ F \text{ is concave along positive directions and convex along cross directions} \]
Properties of Multi-linear Extension

- \( f \) is submodular \( \iff \) \( F \) is DR-submodular

- \( f \) is monotone \( \iff \) \( F \) is monotone

- Multi-linear extension is a linear operator: \( \alpha f_1 + \beta f_2 \iff \alpha F_1 + \beta F_2 \)

- \( F \) is concave along positive directions and convex along cross directions

\[ (v_1, \cdots, v_n), v_i \geq 0 \]
Properties of Multi-linear Extension

\[ f \text{ is submodular} \quad \iff \quad F \text{ is DR-submodular} \]

\[ f \text{ is monotone} \quad \iff \quad F \text{ is monotone} \]

Multi-linear extension is a linear operator:

\[ \alpha f_1 + \beta f_2 \quad \rightarrow \quad \alpha F_1 + \beta F_2 \]

\[ F \text{ is concave along positive directions and convex along cross directions} \]
Properties of Multi-linear Extension

<table>
<thead>
<tr>
<th>$f$ is submodular</th>
<th>$F$ is DR-submodular</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f$ is monotone</td>
<td>$F$ is monotone</td>
</tr>
</tbody>
</table>

Multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \rightarrow \alpha F_1 + \beta F_2$

$F$ is concave along positive directions and convex along cross directions

\[ (v_1, \cdots, v_n), v_i \geq 0 \]  
\[ e_i - e_j \]

\[ \chi \]
Properties of Multi-linear Extension

- $f$ is submodular ↔ $F$ is DR-submodular
- $f$ is monotone ↔ $F$ is monotone

Multi-linear extension is a linear operator: $\alpha f_1 + \beta f_2 \rightarrow \alpha F_1 + \beta F_2$

$F$ is concave along positive directions and convex along cross directions

\[(v_1, \cdots, v_n), v_i \geq 0\]
Let $f$ be a submodular set function with multi-linear extension $F$, then:

$$\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)$$
Let \( f \) be a submodular set function with multi-linear extension \( F \), then:

\[
\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{x \in \mathcal{K}} F(x)
\]
Let $f$ be a submodular set function with multi-linear extension $F$, then:

$$\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)$$

$$\mathcal{I} = \{ S \subseteq V : |S| \leq k \}$$

$$\mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k \}$$

(cardinality)
Let $f$ be a submodular set function with multi-linear extension $F$, then:

\[
\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)
\]

\[\mathcal{I} = \{ S \subseteq V : |S| \leq k \} \quad \rightarrow \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k \} \]

(cardinality)

\[\mathcal{I} = \{ S \subseteq V : \sum_{i \in S} c_i \leq B \} \quad \rightarrow \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} c_i x_i \leq B \} \]

(knapsack)
Let $f$ be a submodular set function with multi-linear extension $F$, then:

\[
\max_{S \in \mathcal{I}} f(S) \quad = \quad \max_{x \in \mathcal{K}} F(x)
\]

\[
\mathcal{I} = \{S \subseteq V : |S| \leq k\} \quad \quad \quad \rightarrow \quad \quad \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k\}
\]

(cardinality)

\[
\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\} \quad \quad \rightarrow \quad \quad \mathcal{K} = \{(x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}
\]

(knapsack)

\[\mathcal{I}\] is a matroid \quad \quad \quad \rightarrow \quad \quad \quad \mathcal{K}\] is the base polytope of the matroid

(matroid)
Let $f$ be a submodular set function with multi-linear extension $F$, then:

$$\max_{S \in \mathcal{I}} f(S) \quad \quad = \quad \quad \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x})$$

\[\mathcal{I} = \{ S \subseteq V : |S| \leq k \} \quad \quad \rightarrow \quad \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k \} \]

(cardinality)

\[\mathcal{I} = \{ S \subseteq V : \sum_{i \in S} c_i \leq B \} \quad \quad \rightarrow \quad \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} c_i x_i \leq B \} \]

(knapsack)

$\mathcal{I}$ is a matroid

$\mathcal{K}$ is the base polytope of the matroid

$\max_{S \in \mathcal{I}} f(S)$
Equivalent Formulations

Let \( f \) be a submodular set function with multi-linear extension \( F \), then:

\[
\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)
\]

\( \mathcal{I} = \{ S \subseteq V : |S| \leq k \} \)

(cardinality)

\( \mathcal{I} \) is a matroid

(matroid)

\( \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k \} \)

\( \mathcal{K} \) is the base polytope of the matroid

(knapsack)
Equivalent Formulations

Let $f$ be a submodular set function with multi-linear extension $F$, then:

$$\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)$$

\[ \mathcal{I} = \{ S \subseteq V : |S| \leq k \} \quad \rightarrow \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k \} \]

\[ \mathcal{I} = \{ S \subseteq V : \sum_{i \in S} c_i \leq B \} \quad \rightarrow \quad \mathcal{K} = \{ (x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} c_i x_i \leq B \} \]

$I$ is a matroid

$\mathcal{K}$ is the base polytope of the matroid
Equivalent Formulations

Let $f$ be a submodular set function with multi-linear extension $F$, then:

\[
\max_{S \in \mathcal{I}} f(S) = \max_{x \in \mathcal{K}} F(x)
\]

- $\mathcal{I} = \{S \subseteq V : |S| \leq k\}$ (cardinality)
- $\mathcal{I} = \{S \subseteq V : \sum_{i \in S} c_i \leq B\}$ (knapsack)
- $\mathcal{I}$ is a matroid

- $\mathcal{K} = \{(x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} x_i \leq k\}$
- $\mathcal{K} = \{(x_1, \cdots, x_n) \in [0, 1]^n : \sum_{i=1}^{n} c_i x_i \leq B\}$
- $\mathcal{K}$ is the base polytope of the matroid

$S_{sol} \in \mathcal{I}$

$x_{sol} \in \mathcal{K}$

rounding
Rounding

$\mathbf{x} = [0, 1]^n$

$\max_{S \in \mathcal{I}} f(S) \quad \rightarrow \quad \max_{x \in \mathcal{K}} F(x)$

$S_{\text{sol}} \in \mathcal{I} \quad \rightarrow \quad x_{\text{sol}} \in \mathcal{K}$

rounding
Rounding

\[ \mathcal{K} = [0, 1]^n \]

\[ \max_{S \in \mathcal{I}} f(S) \quad \rightarrow \quad \max_{x \in \mathcal{K}} F(x) \]

\[ S_{\text{sol}} \in \mathcal{I} \quad \rightarrow \quad x_{\text{sol}} \in \mathcal{K} \]

rounded
Rounding

\[ F \]

\[ x_{\text{sol}} \in \mathcal{K} \]

\[ x_{\text{sol}} = \text{rounding} \]

\[ \mathcal{K} = [0, 1]^n \]

\[ \max_{S \in \mathcal{I}} f(S) \]

\[ \max_{x \in \mathcal{K}} F(x) \]

\[ S_{\text{sol}} \in \mathcal{I} \]

\[ x_{\text{sol}} \in \mathcal{K} \]
Rounding

\[ \mathcal{S} = [0, 1]^n \]

\[ \max_{S \in \mathcal{I}} f(S) \quad \rightarrow \quad \max_{x \in \mathcal{K}} F(x) \]

\[ S_{\text{sol}} \in \mathcal{I} \quad \rightarrow \quad x_{\text{sol}} \in \mathcal{K} \]

\[ \text{rounding} \]
Rounding

\[ F(x) = \begin{cases} 0, & x \in \mathcal{K} \\ 1, & x \notin \mathcal{K} \end{cases} \]

\[ x_{\text{sol}} \in \mathcal{K} \cap \{0, 1\}^n \]

If \( (S)_{\text{max}} \), then \( x_{\text{sol}} \in \mathcal{K} \) is rounded to \( S_{\text{sol}} \in \mathcal{I} \).

\[ \max_{S \in \mathcal{I}} f(S) \quad \rightarrow \quad \max_{x \in \mathcal{K}} F(x) \quad \text{rounding} \quad \mathcal{I} \rightarrow \mathcal{K} \]

\[ S_{\text{sol}} \in \mathcal{I} \quad \leftarrow \quad x_{\text{sol}} \in \mathcal{K} \]
Rounding

\[ \mathcal{X} = [0, 1]^n \]

\( x_{\text{int}} \in \mathcal{K} \cap \{0, 1\}^n \)

\( \rightarrow S_{\text{sol}} \)

\[ \max_{S \in \mathcal{I}} f(S) \]

\( x_{\text{sol}} \in \mathcal{K} \)

\[ \max_{x \in \mathcal{K}} F(x) \]

\( S_{\text{sol}} \in \mathcal{I} \)

\( \text{rounding} \)
Rounding

\[ \mathbf{F} \mathbf{x} = [0, 1]^n \]

If \( \mathbf{F}(\mathbf{x}_{\text{sol}}) \)

\[ \mathbf{x}_{\text{sol}} \in \mathcal{K} \cap \{0, 1\}^n \rightarrow \mathbf{S}_{\text{sol}} \]

\[ \max_{\mathbf{S} \in \mathcal{I}} f(\mathbf{S}) \quad \text{max} \quad \mathbf{F}(\mathbf{x}) \]

\[ \mathbf{S}_{\text{sol}} \in \mathcal{I} \quad \text{rounding} \quad \mathbf{x}_{\text{sol}} \in \mathcal{K} \]
Rounding

\[ F(\mathbf{x}_{\text{sol}}) \]

\[ F(\mathbf{x}_{\text{sol}}) \]

\[ \mathbf{x}_{\text{int}} \in \mathcal{K} \cap \{0, 1\}^n \]

\[ \rightarrow S_{\text{sol}} \]

\[ \max_{S \in \mathcal{I}} f(S) \]

\[ \max_{\mathbf{x} \in \mathcal{K}} F(\mathbf{x}) \]

\[ S_{\text{sol}} \in \mathcal{I} \]

\[ x_{\text{sol}} \in \mathcal{K} \]

rounding
Rounding

\[ f(S_{\text{sol}}) = F(x_{\text{int}}) \geq F(x_{\text{sol}}) \]

\[ x_{\text{int}} \in \mathcal{K} \cap \{0, 1\}^n \rightarrow S_{\text{sol}} \]

\[ \max_{S \in \mathcal{I}} f(S) \rightarrow \max_{x \in \mathcal{K}} F(x) \]

\[ S_{\text{sol}} \in \mathcal{I} \]

\[ x_{\text{sol}} \in \mathcal{K} \]
Several rounding techniques:

- Pipage rounding  \(^{\text{Calinescu et al, 2007, 2011}}\)
- Swap rounding  \(^{\text{Chekuri et al, 2010}}\)
- Contention resolution schemes  \(^{\text{Chekuri et al, 2014}}\)
Pipage Rounding

- Used for Matroid constraints

- We describe the special case of cardinality constraints (for simplicity)
Page Rounding
**Pipage Rounding**

\[
x = (x_1, \cdots, x_n) \in [0, 1]^n
\]

\[
\sum_{i=1}^{n} x_i = k
\]

\[
x_{\text{int}} = (x_1, \cdots, x_n) \in \{0, 1\}^n
\]

\[
\sum_{i=1}^{n} x_i = k
\]
Pipage Rounding

\[ \mathbf{x} = (x_1, \cdots, x_n) \in [0, 1]^n \]

\[ \sum_{i=1}^{n} x_i = k \]

\[ \mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0, 1\}^n \]

\[ \sum_{i=1}^{n} x_i = k \]

while \( \mathbf{x} \) is not integer-valued:

pick two non-integer coordinates \( x_i, x_j \)

if \( x_i + x_j \leq 1 \):

Let \[ \mathbf{x}_1 = \mathbf{x} + x_j(e_i - e_j) \]

\[ \mathbf{x}_2 = \mathbf{x} - x_i(e_i - e_j) \]

if \[ F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \]: \( \mathbf{x} \leftarrow \mathbf{x}_1 \)

else: \( \mathbf{x} \leftarrow \mathbf{x}_2 \)
**Pipage Rounding**

\[ \mathbf{x} = (x_1, \ldots, x_n) \in [0, 1]^n \]

\[ \sum_{i=1}^{n} x_i = k \]

\[ \mathbf{x}_{\text{int}} = (x_1, \ldots, x_n) \in \{0, 1\}^n \]

\[ \sum_{i=1}^{n} x_i = k \]

**while** \( \mathbf{x} \) is not integer-valued:

**pick two non-integer coordinates** \( x_i, x_j \)

**if** \( x_i + x_j \leq 1 \):

Let

\[ \mathbf{x}_1 = \mathbf{x} + x_j (\mathbf{e}_i - \mathbf{e}_j) \]

\[ \mathbf{x}_2 = \mathbf{x} - x_i (\mathbf{e}_i - \mathbf{e}_j) \]

**if** \( F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \): \( \mathbf{x} \leftarrow \mathbf{x}_1 \)

**else:** \( \mathbf{x} \leftarrow \mathbf{x}_2 \)

\( \mathbf{x}_1 = (\ldots, x_i + x_j, \ldots, 0, \ldots) \)

\( \mathbf{x}_2 = (\ldots, 0, \ldots, x_i + x_j, \ldots) \)
Pipage Rounding

\[ \mathbf{x} = (x_1, \cdots, x_n) \in [0, 1]^n \]

\[ \sum_{i=1}^{n} x_i = k \]

\[ \mathbf{x}_{\text{int}} = (x_1, \cdots, x_n) \in \{0, 1\}^n \]

\[ \sum_{i=1}^{n} x_i = k \]

while \( \mathbf{x} \) is not integer-valued:

pick two non-integer coordinates \( x_i, x_j \)

if \( x_i + x_j \leq 1 \):

Let \( \mathbf{x}_1 = \mathbf{x} + x_j (e_i - e_j) \)

Let \( \mathbf{x}_2 = \mathbf{x} - x_i (e_i - e_j) \)

if \( F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \): \( \mathbf{x} \leftarrow \mathbf{x}_1 \)

else: \( \mathbf{x} \leftarrow \mathbf{x}_2 \)

if \( x_i + x_j > 1 \):

Let \( \mathbf{x}_1 = \mathbf{x} + (x_j - 1)(e_i - e_j) \)

Let \( \mathbf{x}_2 = \mathbf{x} - (x_i - 1)(e_i - e_j) \)

if \( F(\mathbf{x}_1) \geq F(\mathbf{x}_2) \): \( \mathbf{x} \leftarrow \mathbf{x}_1 \)

else: \( \mathbf{x} \leftarrow \mathbf{x}_2 \)
Submodular Maximization: Discrete and Continuous

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{Oracle} \]

\[ \max_{x \in \mathcal{K}} F(x) \quad \text{Structure} \]

Perfect

Imperfect

Stochastic

Online

Bandit

Cont. Submodular

DR-submodular

Disc. Submodular
Submodular Maximization: Discrete and Continuous

\[
\max_{S \in \mathcal{I}} f(S) \quad \text{versus} \quad \max_{x \in \mathcal{K}} F(x)
\]

- **Oracle**
  - Perfect
    - Stochastic
  - Imperfect
    - Online
    - Bandit

- **Structure**
  - Cont. Submodular
  - DR-submodular
  - Disc. Submodular
Submodular Maximization: Discrete and Continuous

\[ \max_{S \in \mathcal{I}} f(S) \quad \text{vs} \quad \max_{x \in \mathcal{K}} F(x) \]

Oracle

Perfect

Imperfect

Stochastic

Online

Bandit

Structure

Cont. Submodular

DR-submodular

Disc. Submodular
Stochastic Submodular Maximization: Discrete

\[
\max_{S \in \mathcal{I}} f(S) \doteq \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_\theta(S)]
\]
Stochastic Submodular Maximization: Discrete

\[
\max_{S \in \mathcal{I}} f(S) = \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_{\theta}(S)]
\]
Stochastic Submodular Maximization: Discrete

\[
\max_{S \in \mathcal{I}} f(S) = \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_\theta(S)]
\]

Example: Exemplar Based Clustering

\[
f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)
\]

Data Summarization

\[
f(S) = \mathbb{E}_{H \sim_D [f_H(S)]}
\]

Influence Maximization
Stochastic Submodular Maximization: Discrete

\[
\max_{S \in \mathcal{I}} f(S) \overset{\triangleq}{=} \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_{\theta}(S)]
\]

We see \( f \) through its samples, i.e., \( f_{\theta_1}, f_{\theta_2}, \cdots \)

Data Summarization

\[
f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)
\]

Influence Maximization

\[
f(S) = \mathbb{E}_{H \sim D} [f_H(S)]
\]
Stochastic Submodular Maximization: Discrete

$$\max_{S \in \mathcal{I}} f(S) \triangleq \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D}[f_{\theta}(S)]$$

- We see $f$ through its samples, i.e., $f_{\theta_1}, f_{\theta_2}, \cdots$

Data Summarization

$$f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S)$$

Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$

stochastic oracle

$$S \rightarrow f_{\theta}(S)$$
Stochastic Submodular Maximization: Discrete

\[ \max_{S \in \mathcal{I}} f(S) \overset{!}{=} \max_{S \in \mathcal{I}} \mathbb{E}_{\theta \sim D} [f_{\theta}(S)] \]

- We see \( f \) through its samples, i.e., \( f_{\theta_1}, f_{\theta_2}, \ldots \)

Data Summarization

\[ f(S) = \frac{1}{|V|} \sum_{i \in V} f_i(S) \]

Influence Maximization

\[ f(S) = \mathbb{E}_{H \sim D} [f_H(S)] \]

stochastic oracle

\[ S \quad f_{\theta}(S) \]

\[ \mathbb{E}_{\theta \sim D} [f_{\theta}(S)] = f(S) \]

Yale

Penn
\[ f(S) = \mathbb{E}_{\theta \sim D}[f_{\theta}(S)] \quad \text{multi-linear ext.} \quad F(x) = \mathbb{E}_{\theta \sim D}[F_{\theta}(x)] \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_{\theta}(S)] \quad \text{multi-linear ext.} \quad F(x) = \mathbb{E}_{\theta \sim D}[F_{\theta}(x)] \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

multi-linear ext.

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]
\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

multi-linear ext.

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

\[
\frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})]
\]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \quad \text{multi-linear ext.} \quad F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

\[
\frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})]
\]

unbiased estimate \( g_i \)
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

multi-linear ext.

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

unbiased estimate \( g_i \)

\[
\frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})]
\]

stochastic oracle

\[ [f_\theta(S)] = f(S) \]

\[ S \uparrow \quad f_\theta(S) \downarrow \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

\[ \frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})] \]

unbiased estimate \( g_i \)

stochastic oracle

\[ [f_\theta(S)] = f(S) \]

\[ f_\theta(S) \]

stochastic first order oracle

\[ x \]

\[ g \]

\[ \mathbb{E}[g] = \nabla F(x) \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \quad \text{multi-linear ext.} \quad F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

\[ \frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x} [f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})] \]

unbiased estimate \( g_i \)

\[ g \sim \mathbb{E}[g] = \nabla F(x) \]

stochastic oracle \( f_\theta(S) \)

stochastic first order oracle \( f(S) \)
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

\[ \frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})] \]

\[ \mathbb{E}[g] = \nabla F(x) \]
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_\theta(S)] \]

multi-linear ext.

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_\theta(x)] \]
\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_\theta(x)] \]

unbiased estimate \( g_i \)

\[ \frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_\theta(S \cup \{i\}) - f_\theta(S \setminus \{i\})] \]

stochastic oracle

\[ f_\theta(S) = f(S) \]

\[ \max_{S \in \mathcal{I}} f(S) \]

stochastic first order oracle

\[ x \]
\[ g \]
\[ \mathbb{E}[g] = \nabla F(x) \]

max \( F(x) \)

\[ F(x_{\text{sol}}) \geq (1 - \frac{1}{\epsilon})\text{OPT} - \epsilon \]

SCG
Stochastic Submodular Maximization: Discrete and Continuous

\[ f(S) = \mathbb{E}_{\theta \sim D}[f_{\theta}(S)] \]

multi-linear ext.

\[ F(x) = \mathbb{E}_{\theta \sim D}[F_{\theta}(x)] \]

\[ \nabla F(x) = \mathbb{E}_{\theta \sim D}[\nabla F_{\theta}(x)] \]

unbiased estimate \( g_i \)

\[ \frac{\partial F(x)}{\partial x_i} = \mathbb{E}_{\theta \sim D, S \sim x}[f_{\theta}(S \cup \{i\}) - f_{\theta}(S \setminus \{i\})] \]

stochastic oracle

\[ f(\theta(S)) = f(S) \]

\[ \max_{S \in \mathcal{I}} f(S) \]

maximal \( f(S) \)

stochastic first order oracle

\[ x \]

\[ \max_{x \in \mathcal{K}} F(x) \]

maximal \( F(x) \)

rounding

\[ f(S_{sol}) \geq (1 - \frac{1}{e}) \text{OPT} - \epsilon \]

SCG

\[ F(x_{sol}) \geq (1 - \frac{1}{e}) \text{OPT} - \epsilon \]
Discrete vs. Continuous

Discrete submodular optimization with matroid constraints:

\[ O\left(\frac{n^8}{\varepsilon^4}\right) \rightarrow O\left(\frac{n^2}{\varepsilon^4}\right) \rightarrow O\left(\frac{n^2}{\varepsilon^2}\right) \]

Calinescu et al, 2011  Badanidiyuru, Vondrak, 2013  SCG++, 2019

“Maximizing a monotone submodular function subject to a matroid constraint”, Calinescu, Chekuri, Pal, Vondrák, 2011

“Fast algorithms for maximizing submodular functions”, Badanidiyuru, Vondrak, 2013

“Stochastic Continuous Greedy++: When Upper and Lower Bounds Match”, Hassani, Karbasi, Mokhtari, Shen, 2019
Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:
  \[ O\left(\frac{n^8}{\epsilon^4}\right) \quad \rightarrow \quad O\left(\frac{n^2}{\epsilon^4}\right) \quad \rightarrow \quad O\left(\frac{n^2}{\epsilon^2}\right) \]

- Epinions social network (79k nodes, 580k edges)

Influence Maximization

\[ f(S') = \mathbb{E}_{H \sim D}[f_H(S')] \]
Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:
  $$O\left(\frac{n^8}{\varepsilon^4}\right) \quad \rightarrow \quad O\left(\frac{n^2}{\varepsilon^4}\right) \quad \rightarrow \quad O\left(\frac{n^2}{\varepsilon^2}\right)$$

- Epinions social network (79k nodes, 580k edges)

Influence Maximization

$$f(S) = \mathbb{E}_{H \sim D}[f_H(S)]$$
Discrete vs. Continuous

- Discrete submodular optimization with matroid constraints:
  \[ O\left(\frac{n^8}{\varepsilon^4}\right) \longrightarrow O\left(\frac{n^2}{\varepsilon^4}\right) \longrightarrow O\left(\frac{n^2}{\varepsilon^2}\right) \]

- Epinions social network (79k nodes, 580k edges)

Influence Maximization

\[ f(S) = \mathbb{E}_{H \sim D}[f_H(S)] \]

“Stochastic Submodular Maximization: The Case of Coverage Functions”, Karimi et al.
Submodular Maximization: Discrete and Continuous

$$\max_{S \in \mathcal{I}} f(S)$$

$$\max_{x \in \mathcal{K}} F(x)$$

- **Oracle**
  - Perfect
  - Imperfect
    - Stochastic
    - **Online**
    - Bandit

- **Structure**
  - Disc. Submodular
  - DR-submodular
  - Submodular
Online Submodular Maximization

Online recommendation, advertising, ...
Online Submodular Maximization

Online recommendation, advertising, …
Online Submodular Maximization

Online recommendation, advertising, ...

additional dimension:
Online Submodular Maximization

Online recommendation, advertising, ...

additional dimension: time
Online Submodular Maximization

Online recommendation, advertising, …

additional dimension: $f_1, f_2, f_3, \cdots, f_T$

time
Online Submodular Maximization

Online recommendation, advertising, …

additional dimension: $f_1, f_2, f_3, \ldots, f_T$

time

at the beginning of each round $t$, a set $S_t \in \mathcal{I}$ is selected
Online Submodular Maximization

Online recommendation, advertising, ...

additional dimension: $f_1, f_2, f_3, \cdots, f_T$

- at the beginning of each round $t$, a set $S_t \in \mathcal{I}$ is selected
- at the end of the round, a reward function $f_t$ is revealed
Online Submodular Maximization

Online recommendation, advertising, ...

additional dimension: \( f_1, f_2, f_3, \cdots, f_T \)

time

- at the beginning of each round \( t \), a set \( S_t \in \mathcal{I} \) is selected
- at the end of the round, a reward function \( f_t \) is revealed
- \( f_t : 2^V \to \mathbb{R} \); the reward of round \( t \) is \( f_t(S_t) \)
Online Submodular Maximization

Online recommendation, advertising, ...

additional dimension: \( f_1, f_2, f_3, \ldots, f_T \) vs time

- at the beginning of each round \( t \), a set \( S_t \in \mathcal{I} \) is selected
- at the end of the round, a reward function \( f_t \) is revealed
- \( f_t : 2^V \to \mathbb{R} \); the reward of round \( t \) is \( f_t(S_t) \)

Yale
Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

\[
\text{regret} := \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008
Discrete:

- performance metric:

\[
\text{regret} := \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008
Online Submodular Maximization: Discrete and Continuous

Discrete:

Performance metric:

\[
\text{regret} := \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

best action in hindsight

"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008
Online Submodular Maximization: Discrete and Continuous

Discrete:

- Performance metric:

\[ \text{regret} := \max_{S \in I} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t) \]

- Best action in hindsight
- Algorithm

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008
Online Submodular Maximization: Discrete and Continuous

Discrete:

- **performance metric:**

\[
\alpha - \text{regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

- **online oracle**

\[S_t \in \mathcal{I} \quad f_t(S_t) \quad f_t(\cdot)\]

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008
Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

\[ \alpha - \text{regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t) \]

- best action in hindsight
- algorithm

Continuous:

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018
Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

\[
\begin{align*}
\alpha - \text{regret} &:= \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t) \\
&= \text{best action in hindsight} - \text{algorithm}
\end{align*}
\]

Continuous:

- \(F_1, F_2, \cdots, F_T; F_i : \mathcal{X} \rightarrow \mathbb{R}\)

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018
Online Submodular Maximization: Discrete and Continuous

**Discrete:**

- **Performance metric:**

\[
\alpha - \text{regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

- best action in hindsight

- Algorithm

**Continuous:**

- \( F_1, F_2, \ldots, F_T ; F_i : \mathcal{X} \to \mathbb{R} \)

---

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018
Online Submodular Maximization: Discrete and Continuous

Discrete:
- performance metric:
  \[ \alpha - \text{regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t) \]

Continuous:
- \( F_1, F_2, \cdots, F_T; F_i : \mathcal{X} \to \mathbb{R} \)
- performance metric:
  \[ \alpha - \text{regret} := \alpha \max_{x \in \mathcal{K}} \sum_{t=1}^{T} F_t(x) - \sum_{t=1}^{T} F_t(x_t) \]
Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

\[ \alpha - \text{regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t) \]

\[ \text{best action in hindsight} \]

\[ \text{algorithm} \]

Continuous:

- \( F_1, F_2, \cdots, F_T ; F_i : \mathcal{X} \rightarrow \mathbb{R} \)

Devise algorithms for the online setting with \( \alpha \)-regret being sub-linear in \( T \) best value of \( \alpha \)? lowest regret?
Online Submodular Maximization: Discrete and Continuous

Discrete:

- performance metric:

\[
\alpha \text{-regret} := \alpha \max_{S \in \mathcal{I}} \sum_{t=1}^{T} f_t(S) - \sum_{t=1}^{T} f_t(S_t)
\]

- best action in hindsight

Continuous:

- \( F_1, F_2, \ldots, F_T; F_i : \mathcal{X} \to \mathbb{R} \)

Devise algorithms for the online setting with \( \alpha \)-regret being sub-linear in \( T \) and lowest regret?
Online Gradient Ascent

At each time $t$:

- play $x_t$ and receive reward $F_t(x_t)$

$$x_{t+1} = \text{Proj}_K\{x_t + \eta_t \nabla F_t(x_t)\}$$
At each time $t$:

play $x_t$ and receive reward $F_t(x_t)$

$$x_{t+1} = \text{Proj}_K\{x_t + \eta_t \nabla F_t(x_t)\}$$
At each time $t$:

play $x_t$ and receive reward $F_t(x_t)$

$$x_{t+1} = \text{Proj}_\mathcal{K}\{x_t + \eta_t \nabla F_t(x_t)\}$$
Online Gradient Ascent

At each time \( t \):

- play \( x_t \) and receive reward \( F_t(x_t) \)

\[
x_{t+1} = \text{Proj}_K\{x_t + \eta_t \nabla F_t(x_t)\}
\]

online oracle

\[ x_t \in \mathcal{K} \quad F_t(x_t) \quad F_t(\cdot) \]

If \( F_t \)'s are monotone and DR-submodular, then for the online gradient ascent algorithm we have

\[
\frac{1}{2} - \text{regret} = O(\sqrt{T})
\]

[Chen, Hassani, Karbasi]

“Online Continuous Submodular Maximization”, AISTATS ’18
Online Gradient Ascent

At each time $t$:

- play $x_t$ and receive reward $F_t(x_t)$
- $x_{t+1} = \text{Proj}_\mathcal{K}\{x_t + \eta_t \nabla F_t(x_t)\}$

If $F_t$’s are monotone and DR-submodular, then for the online gradient ascent algorithm we have

$$\frac{1}{2} - \text{regret} = O(\sqrt{T})$$

“Online Continuous Submodular Maximization”, AISTATS ’18

- concave: $F(y) - F(x) \leq \langle \nabla F(x), y - x \rangle$
- monotone DR-submodular: $F(y) - 2F(x) \leq \langle \nabla F(x), y - x \rangle$

[Chen, Hassani, Karbasi]
The Meta-FW Framework:

Franke-Wolfe

regularized follow the leader

play $x_t := x_t^{(K)}$, obtain reward $F_t(x_t)$

$E^{(1)} ightarrow E^{(k+1)} ightarrow E^{(K)}$

action: $v_{t-1}$

feedback: $(v_{t-1}, d_{t-1})$

$x_t^{(k+1)} = (1 - \eta_t)x_t^{(k)} + \eta_t v_t^{(k)}$

$d_t^{(k+1)} = (1 - \rho_t)d_t^{(k)} + \rho_t \nabla f_t(x_t)$

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008

“Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments”, Golovin, Krause, Streeter, 2014

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018

“Projection-Free Online Optimization with Stochastic Gradient: From Convexity to Submodularity”, Chen, Harshaw, Hassani, Karbasi, 2018
Franke-Wolfe

regularized follow the leader

**The Meta-FW Framework:**

play $x_t := x_t^{(K)}$, obtain reward $F_t(x_t)$

Franke-Wolfe

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008

“Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments”, Golovin, Krause, Streeter, 2014

“Online Continuous Submodular Maximization”, Chen, Hassani, Karbasi, 2018

“Projection-Free Online Optimization with Stochastic Gradient: From Convexity to Submodularity”, Chen, Harshaw, Hassani, Karbasi, 2018
The Meta-FW Framework:

If $F_t$’s are monotone and DR-submodular, then for the Meta-Franks-Wolfe algorithm we have

\[
(1 - \frac{1}{e}) - \text{regret} = O(\sqrt{T})
\]
Many open questions:
continuous submodular, non-monotone, other constraints
simpler algorithms, better regret bounds, etc

Related work/ Recent Progress:
The Online Setting

Many open questions:
- continuous submodular, non-monotone, other constraints
- simpler algorithms, better regret bounds, etc

Related work/ Recent Progress:

“Designing smoothing functions for improved worst-case competitive ratio in online optimization”, Eghbali, Fazel, 2016

“No-regret Algos for Online k-submodular optimization”, Soma, 2019

“Consistent Online Optimization: Convex and Submodular”, Karimi, Krause, Lattanzi, Vassilvtiskii, 2019

“Online DR-Submodular Maximization with Stochastic Cumulative Constraints”, Raut, Sadeghi, Fazel, 2020

“Online continuous DR-submodular maximization with long-term budget constraints”, Sadeghi, Fazel, 2020
Submodular Maximization: Discrete and Continuous

\[
\max_{S \in \mathcal{I}} f(S)
\]

\[
\max_{x \in \mathcal{K}} F(x)
\]

---

**Oracle**

- Perfect
- Imperfect
  - Stochastic
  - Online
  - Bandit

**Structure**

- Disc. Submodular
- DR-submodular
- Submodular
Submodular Maximization: Discrete and Continuous

\[
\max_{S \in \mathcal{I}} f(S) \quad \text{vs.} \quad \max_{x \in \mathcal{K}} F(x)
\]

**Oracle**
- Perfect
- Imperfect
  - Stochastic
  - Online
  - Bandit

**Structure**
- Disc. Submodular
- DR-submodular
- Submodular
“Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback, Zhang, Chen, Hassani, Karbasi, 2019

“Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments”, Golovin, Krause, Streeter, 2014

“Linear submodular bandits and their application to diversified retrieval”, Yue, Guestrin, 2012

“An online algorithm for maximizing submodular functions”, Streeter, Golovin, 2008
Bandit Submodular Maximization

If $F_t$'s are monotone and DR-submodular, then for the Bandit Frank-Wolfe algorithm we have

$$(1 - \frac{1}{e}) - \text{regret} = O(T^{8/9})$$

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback", NEURIPS '19

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback, Zhang, Chen, Hassani, Karbasi, 2019

"Online Submodular Maximization under a Matroid Constraint with Application to Learning Assignments", Golovin, Krause, Streeter, 2014

"Linear submodular bandits and their application to diversified retrieval", Yue, Guestrin, 2012

"An online algorithm for maximizing submodular functions", Streeter, Golovin, 2008
Many open questions:
- continuous submodular, non-monotone, other constraints
- simpler algorithms, better regret bounds, etc

If $F_t$’s are monotone and DR-submodular, then for the Bandit Frank-Wolfe algorithm we have

$$(1 - \frac{1}{e}) - \text{regret} = O(T^{\frac{8}{9}})$$

"Online Continuous Submodular Maximization: From Full-Information to Bandit Feedback", NEURIPS ’19
Final Remarks

\[
\max_{S \in \mathcal{I}} f(S) \quad \quad \max_{x \in \mathcal{K}} F(x)
\]

Oracle

Perfect

Imperfect

Stochastic

Online

Bandit

Structure

Disc. Submodular

DR-submodular

Submodular
Final Remarks

\[ \max_{S \in \mathcal{I}} f(S) \]

\[ \max_{x \in \mathcal{K}} F(x) \]

Oracle

Perfect

Imperfect

Stochastic

Online

Bandit

Structure

Disc. Submodular

DR-submodular

Submodular
Final Remarks

- Minimization of continuous submodular functions is another story, see:

  “Submodular Functions: from Discrete to Continuous Domains”, Bach, 2016
Final Remarks

Minimization of continuous submodular functions is another story, see:

“Submodular Functions: from Discrete to Continuous Domains”, Bach, 2016
Final Remarks

Minimization of continuous submodular functions is another story, see:

“Submodular Functions: from Discrete to Continuous Domains”, Bach, 2016

Other continuous extensions of submodular set functions:

\[ F^+ : \text{concave extension} \]

\[ F^- : \text{Lovasz (convex) extension} \]

\[ F : \text{Multi-linear extension} \]
Final Remarks

- Minimization of continuous submodular functions is another story, see:
  
  “Submodular Functions: from Discrete to Continuous Domains”, Bach, 2016

- Other continuous extensions of submodular set functions:

  \[ F^+ : \text{concave extension} \]

  \[ F^- : \text{Lovasz (convex) extension} \]

  \[ F : \text{Multi-linear extension} \]

\[ F^-(x) \leq F(x) \leq F^+(x) \leq (1 - \frac{1}{e})^{-1} F(x) \]

“Submodularity in Combinatorial Optimization”, Vondrak, 2006
Final Remarks

- Minimization of continuous submodular functions is another story, see:
  
  “Submodular Functions: from Discrete to Continuous Domains”, Bach, 2016

- Other continuous extensions of submodular set functions:

  \[ F^+ : \text{concave extension} \]

  \[ F^- : \text{Lovasz (convex) extension} \]

  \[ F : \text{Multi-linear extension} \]

- For coverage and deep submodular functions, tight concave extensions are possible which are also efficiently computable

“Stochastic Submodular Maximization: The Case of coverage Functions”, Karimi, Lucic, Hassani, Krause, 2019

Other Resources on Submodularity

To watch:


(course) J. Bilmes: Submodular Functions, Optimization, & Applications to Machine Learning, University of Washington, 2014 (available on YouTube)

To read (Surveys & Monographs):


N. Buchbinder, M. Feldman: Submodular Functions Maximization Problems, 2018

New: H. Hassani, A. Karbasi: Submodular Optimization: From Discrete to Continuous and Back (in progress, will appear later this year)
Other Resources on Submodularity

- To watch:
  

  (course) J. Bilmes: Submodular Functions, Optimization, & Applications to Machine Learning, University of Washington, 2014 (available on YouTube)

- To read (Surveys & Monographs):


  N. Buchbinder, M. Feldman: Submodular Functions Maximization Problems, 2018

  New: H. Hassani, A. Karbasi: Submodular Optimization: From Discrete to Continuous and Back (in progress, will appear later this year)
Submodular Optimization: From Discrete to Continuous and Back

Hamed Hassani  
Amin Karbasi

Thank you!

Slides + references: http://iid.yale.edu/icml/icml-20.md/