DR-submodularity captures a subclass of non-convex/non-concave functions that enables exact minimization and approximate maximization in poly. time.

- Investigate geometric properties that underlie such objectives, e.g., a strong relation between (approximately) stationary points & global optimum is proved.
- Devise two guaranteed algorithms: 1) A “two-phase” algorithm with ¼ approximation guarantee. 2) A non-monotone Frank-Wolfe variant with ⅜ approximation guarantee.
- Extend to a much broader class of submodular functions on “conic” lattices.

**DR-submodular (Dimensioning Returns) Maximization & Its Applications**

\[ \max_{x \in P} f(x) \]

- If \( f \) is twice differentiable, DR equivalent to \( \max_{x \in \mathbb{R}^n} f(x) \).

**Applications**

- Softmax extension for deterministic point processes [DPPs] [Gillenwater et al ’12]
- Mean-field inference for log-submodular models [Ojiosa et al ’14]
- DR-submodular quadratic programming
- (Generalized submodularity over conic lattices) e.g., logistic regression with a non-convex separable regularizer [Antoniodi et al ’11]
- Etc... (more see paper)

**Two Guaranteed Algorithms**

**NON-MONOTONE FRANK-WOLFE VARIANT**

**Input:** prespecified step size \( \gamma \in (0, 1) \)

While \( \delta \geq 0 \) do:

\[ \psi(k) = \argmin_{t \in [t(k), t(k+1)]} \psi(t) \] /* shrunken LMO */

\[ y_k = \min \{ y_{k-1} (1 - \delta) \} \]

\[ x^{k+1} = x^k + \gamma_k \psi(k), t(k+1) = t_k + \gamma_k, k + 1 \]

**Output:** \( x^0 \)

Guarantee of NON-MONOTONE FRANK-WOLFE VARIANT.

\[ f(x^0) \geq e^{-\gamma} f(x^0) - \frac{(1 - e^{-\gamma})}{2\gamma} f(x^0) - \frac{\delta^2}{\gamma} \]

**Underlying Properties of DR-submodular Maximization**

- Properties based on concavity along non-negative directions:
  - Quadratic Lower Bound: With a 1-Lipschitz gradient, for all \( x \) and \( v \in \mathbb{R}^n \), it holds,
    \[ f(x + \gamma v) \leq f(x) + \gamma f'(x) + \frac{\gamma^2}{2} f''(x) \]
  - Strongly DR-submodular & Quadratic Upper Bound: if for all \( x \) and \( v \in \mathbb{R}^n \), it holds,
    \[ f(x + \gamma v) - f(x) \leq f'(x) + \frac{\gamma^2}{2} f''(x) \]

- Relation Between Approximately Stationary Points & Global Optimum:
  - Lemma. For any two points \( x, y \) coordinate-wise max.
    \[ (y - x)^T f'(x) \leq f(x + y) - f(x) - \frac{\gamma^2}{2} f''(x) \]
  - Proof using concavity along non-negative directions & \( \gamma \)-strong DR-submodularity

**Non-stationarity Measure** *(Generalized from [Lacoste-Julien ’16]).* For any \( q \in X \), the non-stationarity of \( x \) is defined as,

\[ g_q(x) = \max_{z \in X} (x - z)^T f'(x) \]

**Application**

- Based on Local-Global Relation, can use any solver for finding an approximately stationary point as the subroutine, e.g., the Non-convex Frank-Wolfe solver in [Lacoste-Julien ’16]

**Two-Phase Algorithm**

**Input:** stopping tolerance \( \epsilon_1, \epsilon_2 \), iterations \( K_1, K_2 \)

Phase I on \( P \)

- Set \( Q = P \cap \{ y : \| y - u \| \leq \epsilon_1 \} \)
- \( x \leftarrow \) Non-convex Frank-Wolfe (Q, K_1, \epsilon_1)

Phase II on \( Q \)

- Output: \( \argmax_{x \in Q} f(x) \)

**Guarantee of Two-Phase Algorithm.**

\[ \max_{x \in Q} f(x) - \frac{\epsilon_2}{2} \left\| x - x^* \right\|^2 + \frac{\epsilon_1}{2} \left\| x - x^* \right\|^2 + \frac{\epsilon_1}{2} \left\| x - x^* \right\|^2 \]

**Experimental Results** *(more see paper)*

**Baselines:**
- QuadGRiPG: global solver for non-convex quadratically programming (possibly run in exponential time)
- Projected Gradient Ascent (ProcGRAD) with diminishing step sizes (\( \gamma/s_{k+1} \))

**DR-submodular Quadratic Programming**

Synthetic problem instances:

- \( f(x) = \mu A x + k \), \( x \in [-1, 1]^n \)
- \( f(x) = \mu A x + k \), \( x \in [-1, 1]^n \)

Objective and constraints were randomly generated in two manners:

1) Uniform distribution (see Figs below);
2) Exponential distribution

**Maximizing Softmax Extensions for MAP Inference of DPPs.**

\[ f(x) = \log \det (\text{diag}(x) (L - I) + 1), x \in [0, 1]^n \]

- L: kernel/similarity matrix, \( P \) is a matching polytope for matching summarization.

**Synthetic problem instances:**
- Generate Softmax objectives: generate \( n \) eigenvalues \( \lambda \in \mathbb{R}_+ \), each randomly distributed in \([0, 1.5]\).
- Set \( D = \text{diag}(d) \). Generate a random unitary matrix \( U \) and set \( L = UDU^T \).
- Generate polytope constraints similarly as that for quadratic programming

**Real-world results on matched summarization:**

Select a set of document pairs out of a corpus of documents, such that the two documents within a pair are similar, and the overall set of pairs is as diverse as possible. Setting similar to [Gillenwater et al ’12], experimented on the 2012 US Republican debates data.