Overview

- Introduce the notion of curvature, to provide better connections between theory and practice.
- Study the role of curvature in:
  - Approximating submodular functions everywhere
  - Learning Submodular functions
  - Constrained Minimization of submodular functions.
- Provide improved curvature-dependent worst case approximation guarantees and matching hardness results.

Curvature of a Submodular function

- Define three variants of curvature of a monotone submodular function as:
  - \( \kappa_f = 1 - \min_{j \in S} \frac{f(j) - f(j \cup S)}{f(j)} \)
  - \( \kappa(S) = 1 - \min_{j \in S} \frac{f(j \cup S) - f(j)}{f(j)} \)
  - \( \kappa(S) = 1 - \frac{\sum_{j \in S} f(j \cup S) - f(j)}{\sum_{j \in S} f(j)} \)

- Proposition: \( \kappa(S) \leq \kappa_f \leq \kappa(S) \)
- Captures the linearity of a submodular function.
- A more gradual characterization of the hardness of various problems.
- Investigated for submodular maximization (Conforti & Cornuejols, 1984).

Main Ideas

- Curve-Normalized form: Given a monotone submodular function, the curve-normalized version of \( f \) is:
  \[ f^*(X) = \frac{f(X) - (1 - \kappa_f) \sum_{j \in X} f(j)}{\kappa_f} \] 

- Idea: Decompose \( f \) as \( f(X) = f_{\text{omni}}(X) + f_{\text{easy}}(X) \) where \( f_{\text{omni}}(X) = \kappa_f f^*(X) \) and \( f_{\text{easy}}(X) = (1 - \kappa_f) \sum_{j \in X} f(j) \).

- Lemma: If \( f \) is monotone submodular, then \( f^*(X) \) is also monotone non-negative submodular function. Furthermore, \( f^*(X) \leq \sum_{j \in X} f(j) \).

- Lower bounds: Also show curvature-dependent lower bounds.

Approximating Submodular functions Everywhere

Problem: Given a submodular function \( f \) in form of a value oracle, find an approximation \( \hat{f} \) (within polynomial time and space), such that:
\[ f(X) \leq \hat{f}(X) \leq \alpha(n) f(X), \forall X \subseteq V \] for a polynomial \( \alpha(n) \).

We provide a black-box technique to transform bounds into curvature dependent ones.

- Main technique: Approximate the curve-normalized version \( f^* \) as \( \hat{f} \), such that:
\[ \hat{f}(X) \leq f^*(X) \leq \alpha(n) f^*(X) \]

Theorem: The function \( \hat{f}(X) \) satisfies:
\[ \hat{f}(X) \leq \frac{\alpha(n)}{1 + (\alpha(n) - 1)(1 - \kappa_f)} f(X) \leq \frac{1}{1 - \kappa_f} f(X) \]

Ellipsoidal Approximation:
- The Ellipsoidal Approximation algorithm of Goemans et al, provides a function of the form \( \sqrt{w(X)} \) with an approximation factor of \( \alpha(n) = O(\sqrt{\log n}) \).

- Corollary: There exists a function of the form, \( \hat{f}^{\text{ellip}}(X) = \kappa_f \sqrt{w(X)} + (1 - \kappa_f) \sum_{j \in X} f(j) \) such that:
\[ \hat{f}^{\text{ellip}}(X) \leq f(X) \leq O\left(\sqrt{\log n} + (1 + \sqrt{\log n}) (1 - \kappa_f)\right) \]

- Lower bound: Given a submodular function \( f \) with curvature \( \kappa_f \), there does not exist any polynomial-time algorithm that approximates \( f \) within a factor of \( \frac{1}{1 - \kappa_f} \), for any \( \kappa_f > 0 \).

- Modular Upper Bound:
  - A simplest approximation (and upper bound) is \( \hat{f}_M(X) = \sum_{j \in X} f(j) \).
  - Lemma: Given a monotone submodular function \( f \), it holds that:
\[ f(X) \leq \hat{f}_M(X) = \sum_{j \in X} f(j) \leq \frac{|X|}{1 + (|X| - 1)(1 - \kappa_f)} f(X) \]

  This bound is tight for the class of modular approximations.

- Corollary: The class of functions, \( f(X) = \sum_{j \in X} \lambda_j |w(X)|^\alpha \), \( \lambda_j \geq 0 \), satisfies:
\[ f(X) \leq \sum_{j \in X} f(j) \leq |X|^{-\alpha} f(X) \]

Constrained Submodular Minimization

Problem: Minimize a submodular function \( f \) over a family \( \mathcal{C} \) of feasible sets, i.e., \( \min_{X \in \mathcal{C}} f(X) \). \( \mathcal{C} \) could be constraints of the form cardinality (knapsack) constraints, cuts, paths, matchings, trees etc.

- Main framework is to choose a surrogate function \( \hat{f} \), and optimize it instead of \( f \).

- Ellipsoidal Approximation based (EA): Use the curve based Ellipsoidal Approximation as the surrogate function.

- Lemma: For a submodular function with curvature \( \kappa_f < 1 \), algorithm EA will return a solution \( \hat{X} \) that satisfies:
\[ f(\hat{X}) \leq O\left(\frac{\sqrt{\log n}}{(\sqrt{\log n} - 1)(1 - \kappa_f)}\right) f(X) \]

- Modular Upper bound based:
  - Use the simple modular upper bound as a surrogate.
  - Lemma: Let \( X \in \mathcal{C} \) be the solution for minimizing \( \sum_{j \in X} f(j) \) over \( \mathcal{C} \). Then:
\[ f(X) \leq \frac{|X|}{1 + (|X| - 1)(1 - \kappa_f)} f(X) \]

- Corollary: The class of functions, \( f(X) = \sum_{j \in X} \lambda_j |w(X)|^\alpha \), \( \lambda_j \geq 0 \), can be minimized up to a factor of \( |X|^{-\alpha} \).

Constrained Submodular Minimization

Table: Summary of our results for constrained minimization.

<table>
<thead>
<tr>
<th>Constraint</th>
<th>MUB</th>
<th>EA</th>
<th>Curve+Ind.</th>
<th>Lower Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Card. LB</td>
<td>( O(n^{1.5}) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
</tr>
<tr>
<td>Spanning Tree</td>
<td>( O(n^{1.5}) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
</tr>
<tr>
<td>Matchings</td>
<td>( O(n^{1.5}) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
</tr>
<tr>
<td>s-t path</td>
<td>( O(n^{1.5}) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
</tr>
<tr>
<td>s-t cut</td>
<td>( O(n^{1.5}) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
<td>( O(n \log n) )</td>
</tr>
</tbody>
</table>

Effect of Curvature: Polynomial change in the bounds!

- Experiments:
  - Define a function \( f_{\text{ellip}}(X) = n \min\{\kappa(X) R + \beta |X|, \alpha\} \}
  - Choose \( \alpha = n^{1/2} \) and \( \beta = n^2 \), and \( \kappa = \frac{|X|}{|X| - 1} \geq 1 \).

Acknowledgements

Based upon work supported by National Science Foundation Grant No. IIS-1162606, and by a Google, a Microsoft, and an Intel research award. This was also funded in part by Office of Naval Research under grant no. N00014-11-1-0688, NSF CISE Expeditions award CCF-1139158, DARPA XData Award FA8750-12-2-0331, and gifts from Amazon Web Services, Google, SAP, Blue Goji, Cisco, Cleardown Data, Cloudera, Ericsson, Facebook, General Electric, Hortonworks, Intel, Microsoft, NetApp, Oracle, Samsung, Splunk, VMware and Yahoo!