Guarantees for Greedy Maximization of Non-submodular Functions with Applications

ETH zürich

A. An Bian, Joachim M. Buhmann, Andreas Krause and Sebastian Tschiatschek

Problem Setting & Applications

Ground set $\mathcal{V} = \{1, \ldots, n\}$: all "experiments" in experimental design, all variables in continuous programs, all R.V.s in sparse approx. …

Utility function $F(S): 2^{\mathcal{V}} \mapsto \mathbb{R}_+$, monotone ($A \subseteq B \Rightarrow F(A) \leq F(B)$)
But non-submodular/non-supermodular!

Task $\max_{S\subseteq\mathcal{V}, |S|\leq k} F(S)$: select a subset of items with budget k, to maximize the utility $F(S)$

Applications

Class **I** [combinatorial objectives]: Bayesian experimental design [Chaloner '95 ,Krause '08], Sparse Gaussian processes [Lawrence '03], Column subset selection [Altschuler '16] …

Class **II** [auxiliary set fn. in *continuous* opt. with sparsity constraints $\max_{|\mathrm{supp}(\boldsymbol{x})|\leq k} f(\boldsymbol{x})$] $F(S) := \max_{\mathrm{supp}(\boldsymbol{x})\subseteq S} f(\boldsymbol{x}) \to \max_{|S|\leq k} F(S)$:
Feature selection [Guyon '03], Sparse approx. [Das '08, Krause '10, Elenberg '16], Sparse recovery [Candes '03], Sparse M-estimation [Jain '14], LP with combinatorial constraints …

Empirically, Greedy is used for *non-submodular* objectives.

The Greedy Algorithm

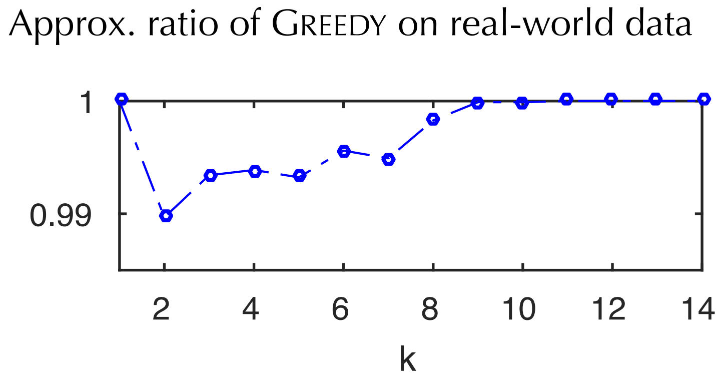
$S^0 \leftarrow \emptyset$
For $t = 1, \ldots, k$ **do**
  $v^* \leftarrow \mathrm{argmax}_{v\in\mathcal{V}\setminus S^{t-1}} \rho_v(S^{t-1})$
  $S^t \leftarrow S^{t-1} \cup \{v^*\}$
Output S^k (Greedy output)

Marginal gain:
$\rho_v(S) := F(S \cup \{v\}) - F(S)$

How Good is Greedy?

Right fig: Bayesian A-optimality $F_A(S)$: reduction of variance in the posterior of parameters.
non-submodular/non-supermodular

Approx. ratio of Greedy on real-world data



Why Greedy is So Good?

First *tight* guarantee for Greedy on k-cardinality non-submodular maximization, *combining* two parameters (α, γ)

Bounding (α, γ) for non-trivial applications

References

Nemhauser, Wolsey, Fisher. An analysis of approximations for maximizing submodular set functions–i. *Mathematical Programming*, 1978.

Conforti, Cornuéjols. Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the rado-edmonds theorem. *Discrete Applied Mathematics*, 1984.

Das, Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. *ICML*, 2011.

Approximation Guarantee

Greedy output

$$F(S^k) \geq \alpha^{-1}\left[1 - \left(\frac{k-\alpha\gamma}{k}\right)^k\right] F(\Omega^*) \geq \alpha^{-1}(1 - e^{-\alpha\gamma})\, F(\Omega^*)$$

optimum

$\alpha \in [0,1]$

Generalized curvature: smallest scalar α s.t. $\forall\, \Omega, S \subseteq \mathcal{V}, i \in S\setminus\Omega$, $\rho_i(S\setminus\{i\} \cup \Omega) \geq (1-\alpha)\rho_i(S\setminus\{i\})$

F is supermodular iff $\alpha = 0$

α How close F is from being supermodular

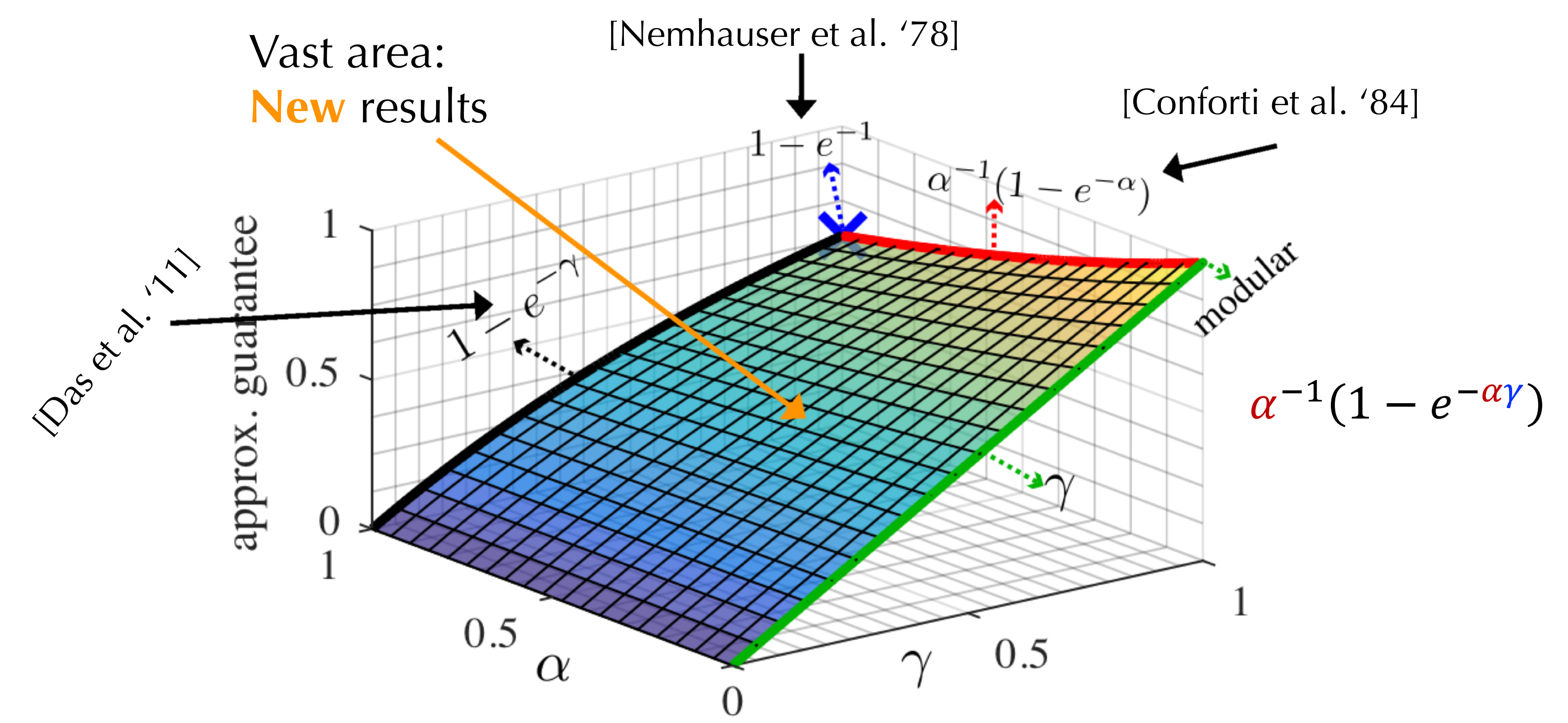
$\gamma \in [0,1]$

Submodularity ratio: [Das et al. '11] largest scalar γ s.t. $\forall\, \Omega, S \subseteq \mathcal{V}$ $\sum_{\omega\in\Omega\setminus S} \rho_\omega(S) \geq \gamma\rho_\Omega(S)$

F is submodular iff $\gamma = 1$

γ To what extent F has submodular property

α and γ can be bounded for non-trivial applications



Corollary: If F is supermodular ($\alpha = 0$, green line above), then approx. guarantee is γ. ($\lim_{\alpha\to 0} \alpha^{-1}(1 - e^{-\alpha\gamma}) = \gamma$)

Tightness Result

$\forall \alpha \in [0,1], \gamma \in (0,1]$, $\exists$ set functions achieving the guarantee exactly

Construction: $\mathcal{V}$ containts elements in $S := \{j_1, \ldots, j_k\}$, $\Omega := \{\omega_1, \ldots, \omega_k\}$ ($S \cap \Omega = \emptyset$), & $n - 2k$ "dummy" elements

$F(T) := \frac{f(|\Omega\cap T|)}{k}\left(1 - \alpha\gamma \sum_{i:j_i\in S\cap T} \xi_i\right) + \sum_{i:j_i\in S\cap T} \xi_i$,

where $\xi_i := \frac{1}{k}\left(\frac{k-\gamma\alpha}{k}\right)^{i-1}$, $i = 1, \ldots, k$, $f(x) := \frac{\gamma^{-1}-1}{k-1}x^2 + \frac{k-\gamma^{-1}}{k-1}x$

$F(T)$: monotone, has curvature α and submodularity ratio γ

Greedy outputs S (proof by induction), optimal solution: Ω
$\frac{F(S)}{F(\Omega)} = \alpha^{-1}\left[1 - \left(\frac{k-\alpha\gamma}{k}\right)^k\right]$ → matching the bound

Bounding α & γ for Applications

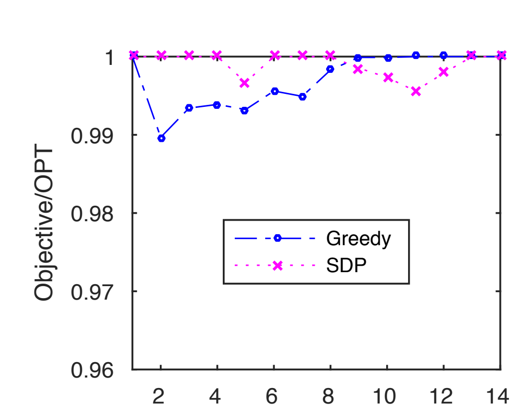
- Bayesian A-optimality: $\boldsymbol{y} = \mathbf{X}^\top\boldsymbol{\theta} + \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2\mathbf{I})$, $\boldsymbol{\theta} \sim \mathcal{N}(0, \beta^{-2}\mathbf{I})$. $F_A(S)$ = const - $\mathrm{tr}\left((\beta^2\mathbf{I} + \sigma^{-2}\mathbf{X}_S\mathbf{X}_S^\top)^{-1}\right)$.
  Assume normalized data $\|\boldsymbol{x}_i\| = 1, \forall i \in \mathcal{V}$, $\|\mathbf{X}\| < \infty$.
  $\gamma \geq \frac{\beta^2}{\|\mathbf{X}\|^2(\beta^2+\sigma^{-2}\|\mathbf{X}\|^2)}$ $\alpha \leq 1 - \frac{\beta^2}{\|\mathbf{X}\|^2(\beta^2+\sigma^{-2}\|\mathbf{X}\|^2)}$
- Determinantal function of a square submatrix: sparse Gaussian process $F(S) = \det(\mathbf{I} + \mathbf{\Sigma}_S)$, $\mathbf{\Sigma}$: covariance matrix. $F(S)$ is supermodular ($\alpha = 0$), γ is lower bounded
- LP with combinatorial constraints, γ is lower bounded

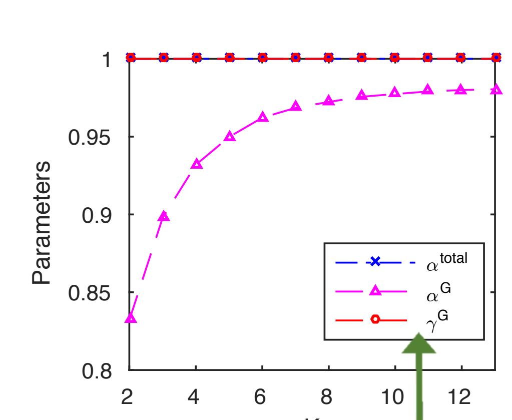
→ Details see paper & source code online

Experiments: Bayesian A-optimality (more see paper)

$\alpha^{\mathrm{total}} := 1 - \min_{i\in\mathcal{V}} \rho_i(\mathcal{V}\setminus\{i\})/\rho_i(\emptyset)$, classical curvature for submodular fn.
less expressive than generalized curvature α

Real-World Results



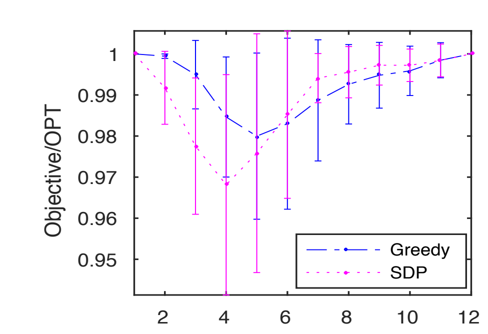


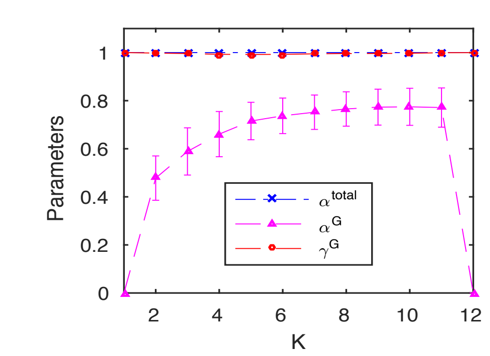
Boston Housing data, n = 14 samples, 14 features
SDP: classical algorithm, but poor scalibility

α^G, γ^G: Greedy/refined version of α, γ. In definitions, restrict $S \to$ Greedy trajectory, $|\Omega| = k$

Synthetic Results

n =12 samples, 6 features, random observations from a multivariate Gaussian with different correlations (0.2 in figs below, 20 repetitions)





Timing

	d: 60 n: 80	d: 40 n: 112	d: 64 n: 128	d: 100 n: 200	d: 120 n: 250
Greedy	0.278	0.360	0.765	4.666	10.56
SDP	95.2	115.2	205.4	1741.2	3883.5
$\frac{\text{SDP}}{\text{Greedy}}$	341.7	319.9	268.7	373.2	367.7

Greedy is 2 orders of magnitude faster than SDP!