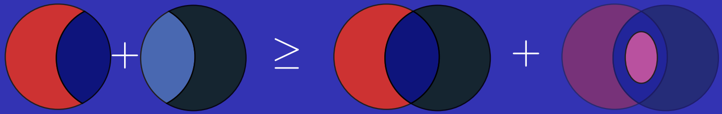

# NOML: Submodularity in Machine Learning

## Friday Laboratory

Jeffrey A. Bilmes and Rishabh Iyer

Departments of Electrical Engineering
& Computer Science and Engineering
University of Washington, Seattle
http://melodi.ee.washington.edu/~bilmes
http://melodi.ee.washington.edu/~rkiyer/

June 17th-19th, 2014

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.
3. Use the similarity scores to define a submodular function $f : 2^V \to \mathbb{R}_+$ (you'll use facility location function, TBD).

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.
3. Use the similarity scores to define a submodular function $f : 2^V \to \mathbb{R}_+$ (you'll use facility location function, TBD).
4. Implement the greedy algorithm to solve $\max_{A \subseteq V, |A| \leq k} f(A)$ and let $A^*$ be the solution, for some small constant integer $k$.

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.
3. Use the similarity scores to define a submodular function $f : 2^V \to \mathbb{R}_+$ (you'll use facility location function, TBD).
4. Implement the greedy algorithm to solve $\max_{A \subseteq V, |A| \leq k} f(A)$ and let $A^*$ be the solution, for some small constant integer $k$.
5. Plot the set of points $V$ (in 2D) and plot the set $A^*$ in a distinct color and see if the points $A^*$ seem to be representative of $V$.

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.
3. Use the similarity scores to define a submodular function $f : 2^V \to \mathbb{R}_+$ (you'll use facility location function, TBD).
4. Implement the greedy algorithm to solve $\max_{A \subseteq V, |A| \leq k} f(A)$ and let $A^*$ be the solution, for some small constant integer $k$.
5. Plot the set of points $V$ (in 2D) and plot the set $A^*$ in a distinct color and see if the points $A^*$ seem to be representative of $V$.
6. Do the above steps repeatedly for different definitions of similarity.

# Submodular Maximization Lab: Goals and Steps

1. Define a random 2D data set of 1000 points $V$ (so $|V| = n = 1000$).
2. Define a non-negative similar score over pairs of points, and then compute those similarities over all pairs. Lets call this $s : V \times V \to \mathbb{R}_+$, so for $v, v' \in V$, $s(v, v')$ is the similarity between $v$ and $v'$.
3. Use the similarity scores to define a submodular function $f : 2^V \to \mathbb{R}_+$ (you'll use facility location function, TBD).
4. Implement the greedy algorithm to solve $\max_{A \subseteq V, |A| \leq k} f(A)$ and let $A^*$ be the solution, for some small constant integer $k$.
5. Plot the set of points $V$ (in 2D) and plot the set $A^*$ in a distinct color and see if the points $A^*$ seem to be representative of $V$.
6. Do the above steps repeatedly for different definitions of similarity.
7. If you have time, randomly partition $V$ into $\ell$ non-empty blocks $V = V_1 \cup V_2 \cup \cdots \cup V_\ell$ and define corresponding small integers $k_1, k_2, \ldots, k_\ell$, $(\forall i, k_i < |V_i|)$ to define a partition matroid $M = (V, \mathcal{I})$, then solve $\max_{A \in \mathcal{I}(M)} f(A)$ and plot the results.

# On the random generation of the $|V| = 1000$ points

- Generate them randomly using any method you want.
- Try to ensure that your final set has at least some obvious outliers.
- You might also want the points to have some sort of natural clusters (that can be seen by plotting the points in 2D).

# Similarity scores over pairs of points in $\mathbb{R}^2$

- Given two points $v, v' \in V$, with $v, v' \in \mathbb{R}^2$, we can define a number of possible similarity scores.
- One option, $\alpha$-parameterized exponentiated $p$-norm, i.e.,:

$$s_{\alpha,p}(v, v') = \exp(-\|v - v'\|_p^p / \alpha) \tag{1}$$

  where $\alpha$ is a scale parameter, and $\|x\|_p = (\sum_i |x_i|^p)^{1/p}$. Try different $\alpha$ & $p$ values to see how the behavior of submodular max changes. Ex: $p = 2$ gives a Gaussian similarity, $p = 1$ gives the 1-norm, and $p = \infty$ gives the max norm.
- Another option doesn't apply the exponential. I.e.,

$$s_a(v, v') = \max_{v_1, v_2 \in V} \|v_1 - v_2\|_2 - \|v - v'\|_2 \tag{2}$$

- Yet another option for similarity: normalized cosine squared. I.e.,

$$s_c(v, v') = \frac{\langle v, v' \rangle^2}{\|v\|_2 \|v'\|_2} \tag{3}$$

  where $\langle v, v' \rangle$ is the dot-product between the two point.

# Facility Location Submodular Function

- Given $s : V \times V \to \mathbb{R}_+$ we define the (uncapacitated) facility location function as $f : 2^V \to \mathbb{R}+$ and for $A \subseteq V$, we have:

$$f(A) = \sum_{v \in V} \max_{a \in A} s(a, v) \tag{4}$$

- Hence, the facility location function is parameterized by the similarity scores you have defined.
- Your goal is to instantiate a number of distinct facility location functions, one each for a given similarity score, and then use it in submodular optimization.

# The greedy algorithm

- Implement the greedy algorithm.

**Algorithm 1:** The Greedy Algorithm

Set $S_0 \leftarrow \emptyset$ ;
**for** $i \leftarrow 0 \ldots (k-1)$ **do**
  Choose $v_i$ as follows:
  $v_i \in \text{argmax}_{v \in V \setminus S_i} f(\{v\}|S_i) = \text{argmax}_{v \in V \setminus S_i} f(S_i \cup \{v\})$ ;
  Set $S_{i+1} \leftarrow S_i \cup \{v_i\}$ ;

- If there are ties in the argmax, break them arbitrarily.

# Plot the points

- Set $k = 50$ and run greedy.
- Neatly and cleanly plot your $V$ points in 2D, and then using a distinct color, indicate the points that the greedy algorithm produced.
- Indicate, in your plot, a number next to each point indicating the order it was chosen by the greedy algorithm.
- Do one separate plot for each different similarity score method. For $s_{\alpha,p}(v, v')$ try at least 10 distinct quite different values of $\alpha$ and $p$.
- Question: what qualitative difference do you find in the result when using the different similarity measures.

# Partition Matroid

- Randomly partition $V$ into $\ell$ non-empty blocks $V = V_1 \cup V_2 \cup \cdots \cup V_\ell$.
- For each block $V_i$, define a small integer $k_i$. Your integer should be small, i.e., $k_i \approx 0.1|V_i|$ at most.
- This gives a set of integers $k_1, k_2, \ldots, k_\ell$.
- We can then define a partition matroid $M = (V, \mathcal{I})$ where the independent sets variable $\mathcal{I}$ is defined in the following way:

$$\mathcal{I} = \{I \subseteq V : |I \cap V_i| \leq k_i, 1 \leq i \leq \ell\} \tag{5}$$

# Constrained submodular max subject to partition matroid constraint

- We can use almost the same greedy algorithm to solve $\max_{A \in \mathcal{I}(M)} f(A)$, where $\mathcal{I}(M)$ is the independent sets of matroid $M$.

**Algorithm 2:** The Greedy Algorithm

Set $S_0 \leftarrow \emptyset$ ;
**for** $i \leftarrow 0 \ldots (|V| - 1)$ **do**
  Choose $v_i$ as follows: $v_i \in \operatorname{argmax}_{v \in V \setminus S_i : S_i \cup \{v\} \in \mathcal{I}(M)} f(\{v\}|S_i)$ ;
  Set $S_{i+1} \leftarrow S_i \cup \{v_i\}$ ;

# Plot the points

- Neatly and cleanly plot your partitioned $V$ points in 2D, where points in the same block are plotted in the same way (e.g., you might plot all points in $V_i$ using the same shape, and points $v \in V_i$ and $v' \in V_j$ for $i \neq j$ are plotted using a different shape).
- Run the matroid-constrained greedy algorithm for each of your similarity measures.
- Indicate the points that the greedy algorithm produced using a distinct color.
- Indicate, in your plot, a number next to each point indicating the order it was chosen by the greedy algorithm.
- Do one separate plot for each different similarity score method. For $s_{\alpha,p}(v, v')$ try at least 10 distinct quite different values of $\alpha$ and $p$.
- Question: what qualitative difference do you find in the result when using the different similarity measures.