10-704: Information Processing and Learning **Fall 2016**

Lecture 4: Sept 12

Lecturer: Aarti Singh

Note: *These notes are based on scribed notes from Spring15 offering of this course. LaTeX template courtesy of UC Berkeley EECS dept.*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

4.1 Maximizing submodular functions

Algorithm 1 Greedy-Algorithm(Ω, f, k)

```
Input: A set $\Omega$, A set function $f : 2^{\Omega} \to \mathbb{R}$, Size of subset $k$.
Output: A subset $A_k \subset \Omega$ of size $k$.
$A_0 \leftarrow \emptyset$
For $i = 1, \dots, k$
    1. for $x \in \Omega \backslash A_{i-1}$, set $\delta_x \leftarrow f(A_{i-1} \cup \{x\}) - f(A_{i-1})$
    2. $x_* \leftarrow \text{argmax}_{x \in \Omega \backslash A_{i-1}} \delta_x$
    3. $A_i \leftarrow A_{i-1} \cup \{x_*\}$
```

Theorem 1 (Nemhauser et al. (1978))**.** *Let f be a function such that:*

1. *f is submodular over finite set Ω*
2. *f is monotone, i.e. $\forall X \subseteq Y \subseteq \Omega$, we have $f(Y) \geq f(X)$*
3. *$f(\emptyset) = 0$*

Let $A_k \subseteq \Omega$ be the first k elements chosen by ***Greedy-Algorithm(Ω, f, k)*** *(see Algorithm 1). Then*

$$f(A_k) \geq \left(1 - \frac{1}{e}\right) f(A_{\text{opt}})$$

where $A_{\text{opt}} = \underset{A \subseteq \Omega, \ \text{card(A)=k}}{\text{argmax}} f(A)$.

Proof. We prove the theorem by induction. Define $A_i = \{a_1, ..., a_i\}$, with $A_0 \equiv \emptyset$. We claim that $\forall 0 \leq j \leq k$

$$f(A_{\text{opt}}) - f(A_j) \leq \left(1 - \frac{1}{k}\right)^j f(A_{\text{opt}}) \tag{4.1}$$

1. At step $j = 0$ we have $f(A_{\text{opt}}) - \underbrace{f(A_0)}_{=f(\emptyset)=0} \leq f(A_{\text{opt}})$

2. Suppose (4.1) is true at step $j = i - 1$. Let $\delta_i = f(A_i) - f(A_{i-1})$. Thus

$$f(A_{\text{opt}}) - f(A_i) = f(A_{\text{opt}}) - f(A_{i-1}) - \delta_i \tag{4.2}$$

Let $A_{\text{opt}} \setminus A_{i-1} = \{x_1, ..., x_m\}, m \leq k$. We have

$$\begin{aligned} f(A_{\text{opt}}) - f(A_{i-1}) &\leq f(A_{\text{opt}} \cup A_{i-1}) - f(A_{i-1}) \quad \text{[monotonicity]} \\ &= f(A_{i-1} \cup (A_{\text{opt}} \setminus A_{i-1})) - f(A_{i-1}) \\ &= \sum_{j=1}^{m} [f(A_{i-1} \cup \{x_1, ..., x_j\}) - f(A_{i-1} \cup \{x_1, ..., x_{j-1}\})] \\ \text{[submodularity]} &\leq \sum_{j=1}^{m} [f(A_{i-1} \cup x_j) - f(A_{i-1})] \\ \text{[choice of } A_i] &\leq \sum_{j=1}^{m} [f(A_i) - f(A_{i-1})] = m\delta_i \leq k\delta_i \end{aligned}$$

$\Rightarrow \delta_i \geq \frac{1}{k}(f(A_{\text{opt}}) - f(A_{i-1}))$. Hence, equation (4.2) can be completed as follows

$$\begin{aligned} f(A_{\text{opt}}) - f(A_i) &= f(A_{\text{opt}}) - f(A_{i-1}) - \delta_i \\ &\leq \left(1 - \frac{1}{k}\right)(f(A_{\text{opt}}) - f(A_{i-1})) \\ &\leq \left(1 - \frac{1}{k}\right)^i f(A_{\text{opt}}) \end{aligned}$$

Therefore (4.1) holds also at step i.

3. Finally notice that $\left(1 - \frac{1}{k}\right)^k \leq \lim_{k \to \infty} = \frac{1}{e}$, which completes the proof.

□

The following theorem works under the more general assumption of "approximate monotonicity" for sets with small cardinality.

Theorem 2 (Krause et al. (2008))**.** *If condition 2 of Theorem 1 is replaced by*

2^*. $\forall X \subseteq \Omega$ *s.t.* $|X| \leq 2k$, *and* $\forall z \in \Omega \setminus X$

$$f(X) \leq f(X \cup \{z\}) + \epsilon \quad \textit{(approximate monotonicity)} \tag{4.3}$$

then $f(A_k) \geq \left(1 - \frac{1}{e}\right)(f(A_{\text{opt}}) - k\epsilon)$.

Clearly notice that if $\epsilon = 0$, then f is monotone. For proof of the theorem see Krause et al. (2008).

Let's check if the assumptions of Theorem 1 are satisfied by entropy $H(X)$ and mutual information $I(X, \Omega \setminus X)$. We have

1. submodularity: $H(X)$ and $I(X, \Omega \setminus X)$ are submodular
2. monotonicity: $H(X) \leq H(Y)$, **but** $I(X, \Omega \setminus X) \not\leq I(Y, \Omega \setminus Y)$

3. $H(\emptyset) = I(\emptyset, \Omega \setminus \emptyset) = 0$

Thus for the mutual information $I(X, \Omega \setminus X)$ Theorem 1 cannot be directly applied. However Theorem 2 can be used under some cases, e.g. if X correspond to a discretization of a Gaussian process with fine enough grid (depending on k and ϵ) - see Lemma 5 and Corollary 6 in Krause et al. (2008).

$$I(X, \Omega \setminus X) \leq I(X \cup \{z\}, \Omega \setminus (X \cup \{z\})) + \epsilon. \tag{4.4}$$

4.2 Differential Entropy

Definition 3 (Differential Entropy)**.** *Let X be a continuous random variable with pdf f. Then the differential entropy of X is defined as*

$$H(X) = -\int f(x) \ln f(x) dx = \mathbb{E}\left[\ln \frac{1}{f(X)}\right]$$

The differential entropy is based on the natural logarithm $\ln = \log_e$, instead of $\log_2$ as for entropy. All of the properties of discrete entropy hold for differential entropy, except the following:

1. The differential entropy can be negative!

Example 1. *$X \sim \text{Uniform}[0, a)$. Then $H(X) = -\int_0^a \frac{1}{a} \ln \frac{1}{a} dx = \ln a$ such that $H(X) < 0, \forall a \in (0, 1)$.*

Example 2. *$X \sim N(0, \sigma^2)$. Then the pdf is $f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$ and*

$$\begin{aligned}
H(X) &= -\int_{\mathbb{R}} f(x) \ln f(x) dx \\
&= -\int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \ln\left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}\right) dx \\
&= \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \left(\frac{x^2}{2\sigma^2} + \ln(\sqrt{2\pi}\sigma)\right) dx \\
&= \frac{1}{2} + \ln(\sqrt{2\pi}\sigma) \\
&= \frac{1}{2} \ln(2\pi e \sigma^2)
\end{aligned}$$

such that $H(X) < 0$ for $\sigma < \sqrt{\frac{1}{2\pi e}}$.

Example 3. *$X \sim N(\mu_{d\times 1}, \Sigma_{d\times d})$. Then $H(X) = \frac{1}{2} \ln((2\pi e)^d |\Sigma|)$.*

4.2.1 Application to Machine Learning: Clustering

Let $X = \{X_1, ..., X_n\}$ be a set of random variables $X_i \in \mathbb{R}^d$. Let $C : X \to \{1, ..., k\}$ define a cluster assignment which put each X_i into one of k classes. Denote with C_i the class assigned to X_i. An *entropy-based clustering* [Faivishevsky et al. (2010)] is performed by solving:

$$\arg\max_C I(X, C) = \arg\min_C H(X|C) \tag{4.5}$$

Connection to k-means clustering

We might wonder if criterion (4.5) differs from k-means clustering. Suppose $P(X|C=j) = N(\mu_j, \sigma_j^2 I)$, where I is the $d \times d$ identity matrix. Thus, the differential entropy of $X|C=j$ is

$$H(X|C=j) = \frac{d}{2}\ln(2\pi e \sigma_j^2) \tag{4.6}$$

An estimator of $H(X|C=j)$ (when μ_j, σ_j are unknown) is the *plug-in* estimator

$$\hat{H}(X|C=j) = \frac{d}{2}\ln\left(2\pi e \frac{1}{n_j}\sum_{i:C_i=j}\|X_i - \hat{\mu}_j\|_2^2\right) \tag{4.7}$$

obtained by replacing σ_j^2 with $\hat{\sigma}_j^2 = \frac{1}{n_j}\sum_{i:C_i=j}\|X_i - \hat{\mu}_j\|^2$, where $\hat{\mu}_j = \frac{1}{n_j}\sum_{i:C_i=j} X_i$ and $n_j = \text{card}(\{i : C_i = j\})$. Thus, an estimate of the conditional (differential) entropy $H(X|C)$ is

$$\hat{H}(X|C) = \sum_{j=1}^{k}\hat{H}(X|C=j)\underbrace{\hat{P}(C=j)}_{n_j/n} = \sum_{j=1}^{k}\frac{d}{2}\ln\left(2\pi e\frac{1}{n_j}\sum_{i:C_i=j}\|X_i-\hat{\mu}_j\|_2^2\right)\frac{n_j}{n} \tag{4.8}$$

Therefore the *entropy-based clustering* (4.5) is implemented by solving

$$\min_C \hat{H}(X|C) = \min_C \sum_{j=1}^{k}\ln\left(\frac{1}{n_j}\sum_{i:C_i=j}\|X_i-\hat{\mu}_j\|_2^2\right)n_j \tag{4.9}$$

The k-means clustering is performed by

$$\min_C \sum_{j=1}^{k}\sum_{i:C_i=j}\|X_i-\hat{\mu}_j\|_2^2 \tag{4.10}$$

We can easily see that the optimization problem of the entropy-based clustering (4.9) differs from the k-means clustering optimization problem (4.10) just because of the logarithm. In fact if ln is replaced by the identity function, (4.9) and (4.10) are equivalent. However, the logarithm makes the entropy-based clustering *more robust* than k-means and minimizes the conditional entropy in each cluster. The entropy based clustering approach also allows for non-Gaussian or even non-parametric assumptions on the distribution of points in each cluster by specifying an appropriate model and estimator for $P(X|C=j)$.

4.3 Estimation of information theoretic quantities

Information theory often assumes the source distribution or probability models are known (as they can often be designed in classical signal processing applications), however in modern signal processing and machine learning applications, these need to be estimated from data i.e. samples from the probability models. We saw an example of this in the clustering application where we estimated the conditional entropy by plugging-in estimates of the mean and variance. Here we revisit that idea and few other estimators. We start by discussing some estimations of entropy in discrete and continuous settings.

4.4 Discrete Setting

We have a distribution P supported on a finite alphabet $\{1, \ldots, d\}$ with $P(X = j) = p_j$ ($\sum_{j=1}^{d} p_j = 1$). We observe independent samples $\{X\}_{i=1}^{n} \sim P$ and would like to estimate some functional of P, say the entropy:

$$H(P) = -\sum_{j=1}^{d} p_j \log_2(p_j) \tag{4.11}$$

4.4.1 Plugin Estimator

The classical estimator here is the **plugin** estimator. We use the sample to estimate the frequencies, $\hat{p}_j = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[X_i = j]$ and simply plug in these frequencies to obtain:

$$\hat{H}_n = -\sum_{j=1}^{d} \hat{p}_j \log_2(\hat{p}_j) \tag{4.12}$$

We will see that $\mathbb{E}\left(\hat{H}_n - H\right)^2 = O\left(\frac{1}{n}\right)$, i.e. the estimator achieves the parametric error rate (error decaying as $1/n$ with number of samples).

References

Krause, Andreas, Singh, A., & Guestrin, C. (2008). Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *The Journal of Machine Learning Research*, *9*, 235-284.

Nemhauser, G. L., Wolsey, L. A., & Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functionsI. *Mathematical Programming*, *14.1*, 265-294.

Faivishevsky, L. & Goldberger, J. (2010). A Nonparametric Information Theoretic Clustering Algorithm. *International Conference on Machine Learning (ICML)*, 2010.