An introduction to Information Theory

Adrish Banerjee

Department of Electrical Engineering
Indian Institute of Technology Kanpur
Kanpur, Uttar Pradesh
India

Sept. 5, 2016

Lecture #14A: Blahut-Arimoto Algorithm

Outline of the lecture

- Alternating Optimization

Outline of the lecture

- Alternating Optimization
- Blahut-Arimoto (BA) algorithm

Outline of the lecture

- Alternating Optimization
- Blahut-Arimoto (BA) algorithm
  - Channel capacity computation

Outline of the lecture

- Alternating Optimization
- Blahut-Arimoto (BA) algorithm
  - Channel capacity computation
  - Rate distortion function computation

Introduction

- For a discrete memoryless channel, p(y/x), the channel capacity is given by

$$C = \max_{r(x)} I(X;Y)$$

where X and Y are respectively the input and output of the channel, and r(x) is the input distribution.

Introduction

- For a discrete memoryless channel, p(y/x), the channel capacity is given by

$$C = \max_{r(x)} I(X;Y)$$

where X and Y are respectively the input and output of the channel, and r(x) is the input distribution.

- The expression for channel capacity is called a single letter characterization in the sense that it depends only on the transition matrix of the channel but not on the blocklength n of the code.

Introduction

- For a discrete memoryless channel, p(y/x), the channel capacity is given by

$$C = \max_{r(x)} I(X;Y)$$

  where X and Y are respectively the input and output of the channel, and r(x) is the input distribution.
- The expression for channel capacity is called a single letter characterization in the sense that it depends only on the transition matrix of the channel but not on the blocklength n of the code.
- When both the input and output alphabet are finite, the computation of channel capacity becomes a finite-dimensional maximization problem.

Introduction

- For a discrete memoryless channel, p(y/x), the channel capacity is given by

$$C = \max_{r(x)} I(X;Y)$$

  where X and Y are respectively the input and output of the channel, and r(x) is the input distribution.
- The expression for channel capacity is called a single letter characterization in the sense that it depends only on the transition matrix of the channel but not on the blocklength n of the code.
- When both the input and output alphabet are finite, the computation of channel capacity becomes a finite-dimensional maximization problem.
- Unless for very special cases, it is not possible to obtain an closed form expression for channel capacity.

Introduction

- For an i.i.d information source $X_k, k \geq 1$ the rate distortion function is given by

$$R(D) = \min_{Q(\hat{x}|x):E(d(x,\hat{x})\leq D)} I(X;\hat{X})$$

where X and $\hat{X}$ are respectively the source and reproduction alphabet, average distortion under single-letter distortion measure d is less than D, $Q(\hat{x}|x)$ is the conditional distribution for which the joint distribution $Q(x,\hat{x})$ satisfies the expected distortion constraint.

Introduction

- For an i.i.d information source $X_k, k \geq 1$ the rate distortion function is given by

$$R(D) = \min_{Q(\hat{x}|x):E(d(x,\hat{x})\leq D)} I(X;\hat{X})$$

where X and $\hat{X}$ are respectively the source and reproduction alphabet, average distortion under single-letter distortion measure d is less than D, $Q(\hat{x}|x)$ is the conditional distribution for which the joint distribution $Q(x,\hat{x})$ satisfies the expected distortion constraint.
- The expression for rate distortion function is also a single letter characterization in the sense that it depends only on the random variable X but not on the blocklength n of the rate distortion code.

Adrish Banerjee Department of Electrical Engineering Indian Institute of Technology Kanpur Kanpur, Uttar Pradesh India
An introduction to Information Theory

Introduction

- For an i.i.d information source $X_k, k \geq 1$ the rate distortion function is given by
$$R(D) = \min_{Q(\hat{x}|x):E(d(x,\hat{x})\leq D)} I(X;\hat{X})$$
where X and $\hat{X}$ are respectively the source and reproduction alphabet, average distortion under single-letter distortion measure d is less than D, $Q(\hat{x}|x)$ is the conditional distribution for which the joint distribution $Q(x,\hat{x})$ satisfies the expected distortion constraint.
- The expression for rate distortion function is also a single letter characterization in the sense that it depends only on the random variable X but not on the blocklength n of the rate distortion code.
- When both the source alphabet and reproduction alphabet are finite, the computation of rate distortion function becomes a finite-dimensional minimization problem.

Introduction

- For an i.i.d information source $X_k, k \geq 1$ the rate distortion function is given by
$$R(D) = \min_{Q(\hat{x}|x):E(d(x,\hat{x})\leq D)} I(X;\hat{X})$$
where X and $\hat{X}$ are respectively the source and reproduction alphabet, average distortion under single-letter distortion measure d is less than D, $Q(\hat{x}|x)$ is the conditional distribution for which the joint distribution $Q(x,\hat{x})$ satisfies the expected distortion constraint.
- The expression for rate distortion function is also a single letter characterization in the sense that it depends only on the random variable X but not on the blocklength n of the rate distortion code.
- When both the source alphabet and reproduction alphabet are finite, the computation of rate distortion function becomes a finite-dimensional minimization problem.
- Unless for very special cases, it is not possible to obtain an closed form expression for rate distortion function, and we have to resort to numerical computation.

Adrish Banerjee Department of Electrical Engineering Indian Institute of Technology Kanpur Kanpur, Uttar Pradesh India
An introduction to Information Theory

Alternating Optimization

- Consider the double supremum

$$\sup_{\mathbf{u_1} \in A_1} \sup_{\mathbf{u_2} \in A_2} f(\mathbf{u_1}, \mathbf{u_2})$$

where A_i is a convex subset of R^{n_i} for $i = 1, 2$ and f is a real function defined on $A_1 \times A_2$. .

Alternating Optimization

- Consider the double supremum

$$\sup_{\mathbf{u_1} \in A_1} \sup_{\mathbf{u_2} \in A_2} f(\mathbf{u_1}, \mathbf{u_2})$$

where A_i is a convex subset of R^{n_i} for $i = 1, 2$ and f is a real function defined on $A_1 \times A_2$. .
- The function f is bounded from above , and is continuous and has continuous partial derivates on $A_1 \times A_2$.

Alternating Optimization

- Consider the double supremum

$$\sup_{\mathbf{u_1}\in A_1}\ \sup_{\mathbf{u_2}\in A_2} f(\mathbf{u_1},\mathbf{u_2})$$

  where A_i is a convex subset of R^{n_i} for $i = 1, 2$ and f is a real function defined on $A_1 \times A_2$. .
- The function f is bounded from above , and is continuous and has continuous partial derivates on $A_1 \times A_2$.
- Assume for all $\mathbf{u_2} \in A_2$, there exists a unique $c_1(\mathbf{u_2}) \in A_1$ such that

$$f(c_1(\mathbf{u_2}),\mathbf{u_2}) = \max_{\mathbf{u_1'}\in A_1} f(\mathbf{u_1'},\mathbf{u_2})$$

Alternating Optimization

- Consider the double supremum

$$\sup_{\mathbf{u_1}\in A_1}\ \sup_{\mathbf{u_2}\in A_2} f(\mathbf{u_1},\mathbf{u_2})$$

  where A_i is a convex subset of R^{n_i} for $i = 1, 2$ and f is a real function defined on $A_1 \times A_2$. .
- The function f is bounded from above , and is continuous and has continuous partial derivates on $A_1 \times A_2$.
- Assume for all $\mathbf{u_2} \in A_2$, there exists a unique $c_1(\mathbf{u_2}) \in A_1$ such that

$$f(c_1(\mathbf{u_2}),\mathbf{u_2}) = \max_{\mathbf{u_1'}\in A_1} f(\mathbf{u_1'},\mathbf{u_2})$$

- Assume for all $\mathbf{u_1} \in A_1$, there exists a unique $c_2(\mathbf{u_1}) \in A_2$ such that

$$f(\mathbf{u_1},c_2(\mathbf{u_1})) = \max_{\mathbf{u_2'}\in A_2} f(\mathbf{u_1},\mathbf{u_2'})$$

Alternating Optimization

- Consider the double supremum

$$\sup_{\mathbf{u_1}\in A_1}\ \sup_{\mathbf{u_2}\in A_2} f(\mathbf{u_1},\mathbf{u_2})$$

  where A_i is a convex subset of R^{n_i} for $i = 1, 2$ and f is a real function defined on $A_1 \times A_2$. .
- The function f is bounded from above , and is continuous and has continuous partial derivates on $A_1 \times A_2$.
- Assume for all $\mathbf{u_2} \in A_2$, there exists a unique $c_1(\mathbf{u_2}) \in A_1$ such that

$$f(c_1(\mathbf{u_2}),\mathbf{u_2}) = \max_{\mathbf{u_1'}\in A_1} f(\mathbf{u_1'},\mathbf{u_2})$$

- Assume for all $\mathbf{u_1} \in A_1$, there exists a unique $c_2(\mathbf{u_1}) \in A_2$ such that

$$f(\mathbf{u_1},c_2(\mathbf{u_1})) = \max_{\mathbf{u_2'}\in A_2} f(\mathbf{u_1},\mathbf{u_2'})$$

- Let $\mathbf{u} = (\mathbf{u_1},\mathbf{u_2})$ and $A_1 \times A_2$. Then the optimization problem is

$$\sup_{\mathbf{u}\in A} f(\mathbf{u})$$

Alternating Optimization

- Let $\mathbf{u}^{(k)} = (\mathbf{u_1}^{(k)},\mathbf{u_2}^{(k)})$ for $k > 0$.

Alternating Optimization

- Let $\mathbf{u}^{(k)} = (\mathbf{u_1}^{(k)}, \mathbf{u_2}^{(k)})$ for $k > 0$.
- Let $\mathbf{u_1}^{(0)}$ be any arbitrary chosen vector in A_1, and let $\mathbf{u_2}^{(0)} = c_1(\mathbf{u_1}^{(0)})$

Alternating Optimization

- Let $\mathbf{u}^{(k)} = (\mathbf{u_1}^{(k)}, \mathbf{u_2}^{(k)})$ for $k > 0$.
- Let $\mathbf{u_1}^{(0)}$ be any arbitrary chosen vector in A_1, and let $\mathbf{u_2}^{(0)} = c_1(\mathbf{u_1}^{(0)})$
- For $k \geq 1$, $\mathbf{u}^{(k)}$ is defined as

$$\begin{aligned} \mathbf{u_1}^{(k)} &= c_1(\mathbf{u_2}^{(k-1)}) \\ \mathbf{u_2}^{(k)} &= c_2(\mathbf{u_1}^{(k)}) \end{aligned}$$

Alternating Optimization

- Let $\mathbf{u}^{(k)} = (\mathbf{u_1}^{(k)}, \mathbf{u_2}^{(k)})$ for $k > 0$.
- Let $\mathbf{u_1}^{(0)}$ be any arbitrary chosen vector in A_1, and let $\mathbf{u_2}^{(0)} = c_1(\mathbf{u_1}^{(0)})$
- For $k \geq 1$, $\mathbf{u}^{(k)}$ is defined as

$$\begin{aligned} \mathbf{u_1}^{(k)} &= c_1(\mathbf{u_2}^{(k-1)}) \\ \mathbf{u_2}^{(k)} &= c_2(\mathbf{u_1}^{(k)}) \end{aligned}$$

- Let the function f at k^{th} iteration $f^{(k)} = f(\mathbf{u}^{(k)})$, then

$$\begin{aligned} f^k &= f(u^k) = f(u_1^k, u_2^k) \\ &\geq f(u_1^k, u_2^{k-1}) \\ &\geq f(u_1^{k-1}, u_2^{k-1}) = f^{k-1} \end{aligned}$$

for $k \geq 1$.

Alternating Optimization

- Since f^k is non decreasing and f is bounded from above, f^k must converge.

Alternating Optimization

- Since f^k is non decreasing and f is bounded from above, f^k must converge.
- Replacing f by -f, the optimization criterion becomes

$$\inf_{\mathbf{u_1} \in A_1} \inf_{\mathbf{u_2} \in A_2} f(\mathbf{u_1}, \mathbf{u_2})$$

  where f is bounded from below.

Alternating Optimization

- Since f^k is non decreasing and f is bounded from above, f^k must converge.
- Replacing f by -f, the optimization criterion becomes

$$\inf_{\mathbf{u_1} \in A_1} \inf_{\mathbf{u_2} \in A_2} f(\mathbf{u_1}, \mathbf{u_2})$$

  where f is bounded from below.
- The double infimum can be computed by the same alternating optimization algorithm.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Let $r(x)p(y|x)$ be a given joint distribution on $X \times Y$ such that $r > 0$, and let q be a transition matrix from Y to X. Then

$$\max_q \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} = \sum_x \sum_y r(x)p(y|x) \log \frac{q^*(x|y)}{r(x)}$$

where the maximization is taken over all q such that $q(x|y) = 0$ if and only if $p(y|x) = 0$ and

$$q^*(x|y) = \frac{r(x)p(y|x)}{\sum_{x'} r(x')p(y|x')}$$

i.e. the maximizing **q** is the one which corresponds to the input distribution **r** and the transition matrix $p(y/x)$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

Proof:

- Let

$$w(y) = \sum_{x'} r(x')p(y|x')$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

Proof:

- Let
$$w(y) = \sum_{x'} r(x')p(y|x')$$
- We assume without loss of generality that for all $y \in Y, p(y/x) > 0$ for some $x \in X$. Since, $\mathbf{r} > 0, w(y) > 0$ for all y, and hence $q^*(x|y)$ is well defined.

Blahut-Arimoto Algorithm: Channel Capacity Computation

Proof:

- Let
$$w(y) = \sum_{x'} r(x')p(y|x')$$
- We assume without loss of generality that for all $y \in Y, p(y/x) > 0$ for some $x \in X$. Since, $\mathbf{r} > 0, w(y) > 0$ for all y, and hence $q^*(x|y)$ is well defined.
- Thus we have
$$r(x)p(y|x) = w(y)q^*(x|y)$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

Proof (Contd.):

- Consider

$$\begin{aligned}
& \sum_x \sum_y r(x)p(y|x) \log \frac{q^*(x|y)}{r(x)} - \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} \\
= & \sum_x \sum_y r(x)p(y|x) \log \frac{q^*(x|y)}{q(x|y)} \\
= & \sum_y \sum_x w(y)q^*(x|y) \log \frac{q^*(x|y)}{q(x|y)} \\
= & \sum_y w(y) \sum_x q^*(x|y) \log \frac{q^*(x|y)}{q(x|y)} \\
= & w(y)D(q^*(x|y)||q(x|y)) \\
\geq & 0
\end{aligned}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- For a discrete memoryless channel $p(y|x)$

$$C = \sup_{\mathbf{r}>0} \max_{\mathbf{q}} \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)}$$

where the maximization is taken over all $\mathbf{q}$ such that $q(x|y) = 0$ if and only if $p(y|x) = 0$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- For a discrete memoryless channel $p(y|x)$

$$C = \sup_{\mathbf{r}>0} \max_{\mathbf{q}} \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)}$$

  where the maximization is taken over all $\mathbf{q}$ such that $q(x|y) = 0$ if and only if $p(y|x) = 0$.
- **Proof:** Let $I(\mathbf{r}, \mathbf{p})$ denote the mutual information $I(X;Y)$ when $\mathbf{r}$ is the input distribution for a channel with transition probability $p(y|x)$. Then

$$C = \max_{\mathbf{r}\geq 0} I(\mathbf{r}, \mathbf{p})$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Let $\mathbf{r}^*$ achieves C. If $\mathbf{r}^* > 0$, then

$$\begin{aligned} C &= \max_{\mathbf{r}\geq \mathbf{0}} I(\mathbf{r}, \mathbf{p}) \\ &= \max_{\mathbf{r}>\mathbf{0}} I(\mathbf{r}, \mathbf{p}) \\ &= \max_{\mathbf{r}>\mathbf{0}} \max_{\mathbf{q}} \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} \\ &= \sup_{\mathbf{r}>\mathbf{0}} \max_{\mathbf{q}} \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} \end{aligned}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Next we consider the case when $\mathbf{r}^* \geq 0$. Since $I(\mathbf{r}, \mathbf{p})$ is continuous in $\mathbf{r}$, for any $\epsilon > 0$, there exists $\delta > 0$, such that if $\|\mathbf{r} - \mathbf{r}^*\| < \delta$, then

$$C - I(\mathbf{r}, \mathbf{p}) < \epsilon$$

where $\|\mathbf{r} - \mathbf{r}^*\|$ denotes the Euclidean distance between $\mathbf{r}$ and $\mathbf{r}^*$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Next we consider the case when $\mathbf{r}^* \geq 0$. Since $I(\mathbf{r}, \mathbf{p})$ is continuous in $\mathbf{r}$, for any $\epsilon > 0$, there exists $\delta > 0$, such that if $\|\mathbf{r} - \mathbf{r}^*\| < \delta$, then

$$C - I(\mathbf{r}, \mathbf{p}) < \epsilon$$

where $\|\mathbf{r} - \mathbf{r}^*\|$ denotes the Euclidean distance between $\mathbf{r}$ and $\mathbf{r}^*$.

- In particular, there exists $\tilde{\mathbf{r}} > 0$ that satisfies the above equation, then

$$\begin{aligned} C &= \max_{\mathbf{r} \geq 0} I(\mathbf{r}, \mathbf{p}) \\ &\geq \sup_{\mathbf{r} > 0} I(\mathbf{r}, \mathbf{p}) \\ &\geq I(\tilde{\mathbf{r}}, p) \\ &> C - \epsilon \end{aligned}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Thus we have

$$C - \epsilon < \sup_{\mathbf{r}>0} I(\mathbf{r}, \mathbf{p}) \leq C$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- Thus we have

$$C - \epsilon < \sup_{\mathbf{r}>0} I(\mathbf{r}, \mathbf{p}) \leq C$$

- By letting $\epsilon \to 0$, we conclude

$$C = \sup_{\mathbf{r}>0} I(\mathbf{r}, \mathbf{p}) = \sup_{\mathbf{r}>0} \max_{\mathbf{q}} \sum_x \sum_y r(x) p(y|x) \log \frac{q(x|y)}{r(x)}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use alternating optimization algorithm to compute capacity.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use alternating optimization algorithm to compute capacity.
- We arbitrary choose a strictly positive input distribution in A_1 and let it be $\mathbf{r}^{(0)}$. We define $\mathbf{q}^{(0)}$ and in general $\mathbf{q}^{(k)}$ for $k \geq 0$.

$$q^{(k)}(x|y) = \frac{r^{(k)}(x)p(y|x)}{\sum_{x'} r^{(k)}(x')p(y|x')}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use alternating optimization algorithm to compute capacity.
- We arbitrary choose a strictly positive input distribution in A_1 and let it be $\mathbf{r}^{(0)}$. We define $\mathbf{q}^{(0)}$ and in general $\mathbf{q}^{(k)}$ for $k \geq 0$.

$$q^{(k)}(x|y) = \frac{r^{(k)}(x)p(y|x)}{\sum_{x'} r^{(k)}(x')p(y|x')}$$

- In order to define $\mathbf{r}^{(k)}$ for $k \geq 1$, we need to find $\mathbf{r} \in A_1$ that maximizes the function for a given $\mathbf{q} \in A_2$, where the constraints on $\mathbf{r}$ are $\sum_x r(x) = 1$ and $r(x) > 0$ for all $x \in X$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use the method of Lagrange multipliers to find the best r. Ignoring temporarily the positivity constraints on $\mathbf{r}$, we get

$$J = \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} - \lambda \sum_x r(x)$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use the method of Lagrange multipliers to find the best r. Ignoring temporarily the positivity constraints on **r**, we get

$$J = \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} - \lambda \sum_x r(x)$$

- Differentiating with respect to r(x), we get

$$\frac{\partial J}{\partial r(x)} = \sum_y p(y|x) \log q(x|y) - \log r(x) - 1 - \lambda$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We use the method of Lagrange multipliers to find the best r. Ignoring temporarily the positivity constraints on **r**, we get

$$J = \sum_x \sum_y r(x)p(y|x) \log \frac{q(x|y)}{r(x)} - \lambda \sum_x r(x)$$

- Differentiating with respect to r(x), we get

$$\frac{\partial J}{\partial r(x)} = \sum_y p(y|x) \log q(x|y) - \log r(x) - 1 - \lambda$$

- Equating $\frac{\partial J}{\partial r(x)}$ to zero, we get

$$\log r(x) = \sum_y p(y|x) \log q(x|y) - 1 - \lambda$$

or

$$r(x) = e^{-(\lambda+1)} \prod_y q(x|y)^{p(y|x)}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We know that $\sum r(x) = 1$, hence

$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_{x'} \prod_y q(x'|y)^{p(y|x')}}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We know that $\sum r(x) = 1$, hence

$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_{x'} \prod_y q(x'|y)^{p(y|x')}}$$

- The product is over all y such that $p(y|x) > 0$ and $q(x|y) > 0$ for all such y.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We know that $\sum r(x) = 1$, hence

$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_{x'} \prod_y q(x'|y)^{p(y|x')}}$$

- The product is over all y such that $p(y|x) > 0$ and $q(x|y) > 0$ for all such y.
- This implies that both numerator and denominator terms on the right hand side above are positive and hence, $r(x) > 0$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- We know that $\sum r(x) = 1$, hence

$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_{x'} \prod_y q(x'|y)^{p(y|x')}}$$

- The product is over all y such that $p(y|x) > 0$ and $q(x|y) > 0$ for all such y.
- This implies that both numerator and denominator terms on the right hand side above are positive and hence, $r(x) > 0$.
- Hence we define $r^{(k)}$ for $k \geq 1$

$$r^{(k)}(x) = \frac{\prod_y q^{(k-1)}(x|y)^{p(y|x)}}{\sum_{x'} \prod_y q^{(k-1)}(x'|y)^{p(y|x')}}$$

Blahut-Arimoto Algorithm: Channel Capacity Computation

- The vectors $\mathbf{r}^{(k)}$ and $\mathbf{q}^{(k)}$ are defined in the order $\mathbf{r}^{(0)}, \mathbf{q}^{(0)}, \mathbf{r}^{(1)}, \mathbf{q}^{(1)}, \cdots$ where each vector in the sequence is a function of the previous vector, except $\mathbf{r}^{(0)}$ that is chosen arbitrarily in A_1.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- The vectors $\mathbf{r}^{(k)}$ and $\mathbf{q}^{(k)}$ are defined in the order $\mathbf{r}^{(0)}, \mathbf{q}^{(0)}, \mathbf{r}^{(1)}, \mathbf{q}^{(1)}, \cdots$ where each vector in the sequence is a function of the previous vector, except $\mathbf{r}^{(0)}$ that is chosen arbitrarily in A_1.
- It can be shown by mathematical induction that $\mathbf{r}^{(k)} \in A_1$ and $\mathbf{q}^{(k)} \in A_2$ for all $k \geq 0$.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- The vectors $\mathbf{r}^{(k)}$ and $\mathbf{q}^{(k)}$ are defined in the order $\mathbf{r}^{(0)}, \mathbf{q}^{(0)}, \mathbf{r}^{(1)}, \mathbf{q}^{(1)}, \cdots$ where each vector in the sequence is a function of the previous vector, except $\mathbf{r}^{(0)}$ that is chosen arbitrarily in A_1.
- It can be shown by mathematical induction that $\mathbf{r}^{(k)} \in A_1$ and $\mathbf{q}^{(k)} \in A_2$ for all $k \geq 0$.
- Upon determining $\mathbf{r}^{(k)}, \mathbf{q}^{(k)}$, we compute $f^{(k)} = f(\mathbf{r}^{(k)}, \mathbf{q}^{(k)})$ for all k.

Blahut-Arimoto Algorithm: Channel Capacity Computation

- The vectors $\mathbf{r}^{(k)}$ and $\mathbf{q}^{(k)}$ are defined in the order $\mathbf{r}^{(0)}, \mathbf{q}^{(0)}, \mathbf{r}^{(1)}, \mathbf{q}^{(1)}, \cdots$ where each vector in the sequence is a function of the previous vector, except $\mathbf{r}^{(0)}$ that is chosen arbitrarily in A_1.
- It can be shown by mathematical induction that $\mathbf{r}^{(k)} \in A_1$ and $\mathbf{q}^{(k)} \in A_2$ for all $k \geq 0$.
- Upon determining $\mathbf{r}^{(k)}, \mathbf{q}^{(k)}$, we compute $f^{(k)} = f(\mathbf{r}^{(k)}, \mathbf{q}^{(k)})$ for all k.
- If f is a concave function, $f^{(k)} \rightarrow C$

BA Algorithm: Convergence

- In order to show that Blahut Arimoto algorithm for computing channel capacity converges, we need to show that the function

$$f(\mathbf{r},\mathbf{q}) = \sum_x \sum_y r(x)p(y/x)\log\frac{q(x|y)}{r(x)}$$

is concave.

BA Algorithm: Convergence

- In order to show that Blahut Arimoto algorithm for computing channel capacity converges, we need to show that the function

$$f(\mathbf{r},\mathbf{q}) = \sum_x \sum_y r(x)p(y/x)\log\frac{q(x|y)}{r(x)}$$

is concave.

- Let us consider two ordered pairs $(\mathbf{r_1},\mathbf{q_1})$ and $(\mathbf{r_2},\mathbf{q_2})$. For any $0 \leq \lambda \leq 1$, we have using log sum inequality

$$\begin{aligned} & (\lambda r_1(x) + (1-\lambda)r_2(x))\log\frac{\lambda r_1(x) + (1-\lambda)r_2(x)}{\lambda q_1(x|y) + (1-\lambda)q_2(x|y)} \\ \leq\ & \lambda r_1(x)\log\frac{r_1(x)}{q_1(x|y)} + (1-\lambda)r_2(x)\log\frac{r_2(x)}{q_2(x|y)} \end{aligned}$$

BA Algorithm: Convergence

- Taking the reciprocal of the logarithms, we get

$$\begin{aligned} & (\lambda r_1(x) + (1-\lambda) r_2(x)) \log \frac{\lambda q_1(x|y) + (1-\lambda) q_2(x|y)}{\lambda r_1(x) + (1-\lambda) r_2(x)} \\ \geq \quad & \lambda r_1(x) \log \frac{q_1(x|y)}{r_1(x)} + (1-\lambda) r_2(x) \log \frac{q_2(x|y)}{r_2(x)} \end{aligned}$$

BA Algorithm: Convergence

- Taking the reciprocal of the logarithms, we get

$$\begin{aligned} & (\lambda r_1(x) + (1-\lambda) r_2(x)) \log \frac{\lambda q_1(x|y) + (1-\lambda) q_2(x|y)}{\lambda r_1(x) + (1-\lambda) r_2(x)} \\ \geq \quad & \lambda r_1(x) \log \frac{q_1(x|y)}{r_1(x)} + (1-\lambda) r_2(x) \log \frac{q_2(x|y)}{r_2(x)} \end{aligned}$$

- Multiplying both sides by $p(y|x)$ and summing over all x and y, we get

$$f(\lambda \mathbf{r}_1 + (1-\lambda)\mathbf{r}_2, \lambda \mathbf{q}_1 + (1-\lambda)\mathbf{q}_2) \geq \lambda f(\mathbf{r}_1, \mathbf{q}_1) + (1-\lambda) f(\mathbf{r}_2, \mathbf{q}_2)$$

BA Algorithm: Convergence

- Taking the reciprocal of the logarithms, we get

$$\begin{aligned}
& (\lambda r_1(x) + (1-\lambda) r_2(x)) \log \frac{\lambda q_1(x|y) + (1-\lambda) q_2(x|y)}{\lambda r_1(x) + (1-\lambda) r_2(x)} \\
\geq \quad & \lambda r_1(x) \log \frac{q_1(x|y)}{r_1(x)} + (1-\lambda) r_2(x) \log \frac{q_2(x|y)}{r_2(x)}
\end{aligned}$$

- Multiplying both sides by $p(y|x)$ and summing over all x and y, we get

$$f(\lambda \mathbf{r}_1 + (1-\lambda)\mathbf{r}_2, \lambda \mathbf{q}_1 + (1-\lambda)\mathbf{q}_2) \geq \lambda f(\mathbf{r}_1, \mathbf{q}_1) + (1-\lambda) f(\mathbf{r}_2, \mathbf{q}_2)$$

- Therefore f is concave.

BA Algorithm: Rate Distortion Function Computation

- For all points of interest, $R(0) > 0$, otherwise $R(D) = 0$ for all $D \geq 0$.

BA Algorithm: Rate Distortion Function Computation

- For all points of interest, $R(0) > 0$, otherwise $R(D) = 0$ for all $D \geq 0$.
- Also, R(D) is strictly decreasing for $0 \leq D \leq D_{\text{max}}$.

BA Algorithm: Rate Distortion Function Computation

- For all points of interest, $R(0) > 0$, otherwise $R(D) = 0$ for all $D \geq 0$.
- Also, R(D) is strictly decreasing for $0 \leq D \leq D_{\text{max}}$.
- Since, R(D) is convex, for any $s \leq 0$, there exists a point on R(D) curve for $0 \leq D \leq D_{\text{max}}$ such that the slope of a tangent to the R(D) curve at that point is equal to s. Denote such a point on the R(D) curve by $(D_s, R(D_s))$

BA Algorithm: Rate Distortion Function Computation

- For all points of interest, $R(0) > 0$, otherwise $R(D) = 0$ for all $D \geq 0$.
- Also, R(D) is strictly decreasing for $0 \leq D \leq D_{\max}$.
- Since, R(D) is convex, for any $s \leq 0$, there exists a point on R(D) curve for $0 \leq D \leq D_{\max}$ such that the slope of a tangent to the R(D) curve at that point is equal to s. Denote such a point on the R(D) curve by $(D_s, R(D_s))$
- For $s \leq 0$, the tangent to the rate-distortion function R(D) at $(D_s, R(D_s))$ has slope s and intersects with the ordinate at $R(D_s) - sD_s$.

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.
- Then for any $\mathbf{Q}$, $(I(\mathbf{p}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ is a point in the rate distortion region, and the line with slope s passing through $(I(\mathbf{p}, \mathbf{Q}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ interests the ordinate at $I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})$

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.
- Then for any $\mathbf{Q}$, $(I(\mathbf{p}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ is a point in the rate distortion region, and the line with slope s passing through $(I(\mathbf{p}, \mathbf{Q}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ interests the ordinate at $I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})$
- Since the R(D) curve defines the boundary of the rate-distortion region, we see that

$$R(D_s) - sD_s = \min_{\mathbf{Q}}[I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})]$$

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.
- Then for any $\mathbf{Q}$, $(I(\mathbf{p}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ is a point in the rate distortion region, and the line with slope s passing through $(I(\mathbf{p}, \mathbf{Q}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ interests the ordinate at $I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})$
- Since the R(D) curve defines the boundary of the rate-distortion region, we see that

$$R(D_s) - sD_s = \min_{\mathbf{Q}}[I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})]$$

- For each $s \leq 0$, if we can find a $\mathbf{Q_s}$ that achieves the minimum, then the line passing through $(0, I(\mathbf{p}, \mathbf{Q_s}) - sD(\mathbf{p}, \mathbf{Q_s}))$ gives a tight lower bound on the R(D) curve.

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.
- Then for any $\mathbf{Q}$, $(I(\mathbf{p}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ is a point in the rate distortion region, and the line with slope s passing through $(I(\mathbf{p}, \mathbf{Q}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ interests the ordinate at $I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})$
- Since the R(D) curve defines the boundary of the rate-distortion region, we see that

$$R(D_s) - sD_s = \min_{\mathbf{Q}}[I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})]$$

- For each $s \leq 0$, if we can find a $\mathbf{Q_s}$ that achieves the minimum, then the line passing through $(0, I(\mathbf{p}, \mathbf{Q_s}) - sD(\mathbf{p}, \mathbf{Q_s}))$ gives a tight lower bound on the R(D) curve.
- In particular, if $(R(D_s), D_s)$ is unique, then $D_s = D(\mathbf{p}, \mathbf{Q_s})$ and $R(D_s) = I(\mathbf{p}, \mathbf{Q_s})$.

BA Algorithm: Rate Distortion Function Computation

- Let $I(\mathbf{p}, \mathbf{Q})$ denote the mutual information $I(X, \hat{X})$ and $D(\mathbf{p}, \mathbf{Q})$ denote the expected distortion $Ed(X, \hat{X})$ when $\mathbf{p}$ is the distribution for X and $\mathbf{Q}$ is the transition matrix from X to $\hat{X}$.
- Then for any $\mathbf{Q}$, $(I(\mathbf{p}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ is a point in the rate distortion region, and the line with slope s passing through $(I(\mathbf{p}, \mathbf{Q}, \mathbf{Q}), D(\mathbf{p}, \mathbf{Q}))$ interests the ordinate at $I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})$
- Since the R(D) curve defines the boundary of the rate-distortion region, we see that

$$R(D_s) - sD_s = \min_{\mathbf{Q}}[I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})]$$

- For each $s \leq 0$, if we can find a $\mathbf{Q_s}$ that achieves the minimum, then the line passing through $(0, I(\mathbf{p}, \mathbf{Q_s}) - sD(\mathbf{p}, \mathbf{Q_s}))$ gives a tight lower bound on the R(D) curve.
- In particular, if $(R(D_s), D_s)$ is unique, then $D_s = D(\mathbf{p}, \mathbf{Q_s})$ and $R(D_s) = I(\mathbf{p}, \mathbf{Q_s})$.
- By varying over all $s \leq 0$, we can trace out the whole R(D) curve.

BA Algorithm: Rate Distortion Function Computation

- Let $p(x)Q(\hat{x}|x)$ be a given joint distribution on $X \times \hat{X}$ such that $\mathbf{Q} > 0$, and let $\mathbf{t}$ be any distribution on $\hat{X}$ such that $\mathbf{t} > 0$. Then

$$\min_{\mathbf{t}>0} \sum_x \sum_{\hat{x}} p(x)Q(\hat{x}|x) \log \frac{Q(\hat{x}|x)}{t(\hat{x})} = \sum_x \sum_{\hat{x}} p(x)Q(\hat{x}|x) \log \frac{Q(\hat{x}|x)}{t^*(\hat{x})}$$

where $t^*(\hat{x}) = \sum_x p(x)Q(\hat{x}|x)$

BA Algorithm: Rate Distortion Function Computation

- Let $p(x)Q(\hat{x}|x)$ be a given joint distribution on $X \times \hat{X}$ such that $\mathbf{Q} > 0$, and let $\mathbf{t}$ be any distribution on $\hat{X}$ such that $\mathbf{t} > 0$. Then

$$\min_{\mathbf{t}>0} \sum_x \sum_{\hat{x}} p(x)Q(\hat{x}|x) \log \frac{Q(\hat{x}|x)}{t(\hat{x})} = \sum_x \sum_{\hat{x}} p(x)Q(\hat{x}|x) \log \frac{Q(\hat{x}|x)}{t^*(\hat{x})}$$

  where $t^*(\hat{x}) = \sum_x p(x)Q(\hat{x}|x)$
- Applying the above lemma, we can write

$$R(D_s) - sD_s = \min_{\mathbf{Q}}[I(\mathbf{p}, \mathbf{Q}) - sD(\mathbf{p}, \mathbf{Q})]$$

$$= \inf_{Q>0} \min_{t>0} \left[\sum_{x,\hat{x}} p(x)Q(\hat{x}|x) \log \frac{Q(\hat{x}/x)}{t(\hat{x})} - s \sum_{x,\hat{x}} p(x)Q(\hat{x}|x)d(x,\hat{x}) \right]$$

BA Algorithm: Rate Distortion Function Computation

- We can now apply alternating optimization algorithm.

Adrish Banerjee Department of Electrical Engineering Indian Institute of Technology Kanpur Kanpur, Uttar Pradesh India
An introduction to Information Theory

BA Algorithm: Rate Distortion Function Computation

- We can now apply alternating optimization algorithm.
- For computation of rate-distortion function, we start with any strictly positive transition matrix $\mathbf{Q}^{(0)}$.

BA Algorithm: Rate Distortion Function Computation

- We can now apply alternating optimization algorithm.
- For computation of rate-distortion function, we start with any strictly positive transition matrix $\mathbf{Q}^{(0)}$.
- Then we define $\mathbf{t}^{(0)}$ and in general $\mathbf{t}^{(\mathbf{k})}$ as

$$t^{(k)}(\hat{x}) = \sum_{x} p(x) Q^{(k)}(\hat{x}|x)$$

BA Algorithm: Rate Distortion Function Computation

- We can now apply alternating optimization algorithm.
- For computation of rate-distortion function, we start with any strictly positive transition matrix $\mathbf{Q}^{(0)}$.
- Then we define $\mathbf{t}^{(0)}$ and in general $\mathbf{t}^{(\mathbf{k})}$ as

$$t^{(k)}(\hat{x}) = \sum_x p(x) Q^{(k)}(\hat{x}|x)$$

- In order to define $\mathbf{Q}^{(1)}$, and in general $\mathbf{Q}^{(k)}$, we need to find $\mathbf{Q}$ that minimizes the function for a given $\mathbf{t}$, where the constraints on $\mathbf{Q}$ are

$$Q(\hat{x}|x) > 0$$

for all $(x, \hat{x}) \in X \times \hat{X}$ and

$$\sum_{\hat{x}} Q(\hat{x}|x) = 1$$

for all $x \in X$.

BA Algorithm: Rate Distortion Function Computation

- Following the same procedure as in computation of channel capacity, we get

$$Q^{(k)}(\hat{x}|x) = \frac{t^{(k-1)}(\hat{x}) e^{sd(x,\hat{x})}}{\sum_{\hat{x}'} t^{(k-1)}(\hat{x}') e^{sd(x,\hat{x}')}}$$

BA Algorithm: Rate Distortion Function Computation

- Following the same procedure as in computation of channel capacity, we get

$$Q^{(k)}(\hat{x}|x) = \frac{t^{(k-1)}(\hat{x})e^{sd(x,\hat{x})}}{\sum_{\hat{x}'} t^{(k-1)}(\hat{x}')e^{sd(x,\hat{x}')}}$$

- If there exists a unique point $(R(D_s), D_s)$ on the R(D) curve such that the slope of the tangent at that point is equal to s, then

$$(I(\mathbf{p}, \mathbf{Q}^{(k)}), D(\mathbf{p}, \mathbf{Q}^{(k)})) \to (R(D_s), D_s)$$

BA Algorithm: Rate Distortion Function Computation

- Following the same procedure as in computation of channel capacity, we get

$$Q^{(k)}(\hat{x}|x) = \frac{t^{(k-1)}(\hat{x})e^{sd(x,\hat{x})}}{\sum_{\hat{x}'} t^{(k-1)}(\hat{x}')e^{sd(x,\hat{x}')}}$$

- If there exists a unique point $(R(D_s), D_s)$ on the R(D) curve such that the slope of the tangent at that point is equal to s, then

$$(I(\mathbf{p}, \mathbf{Q}^{(k)}), D(\mathbf{p}, \mathbf{Q}^{(k)})) \to (R(D_s), D_s)$$

- Otherwise $(I(\mathbf{p}, \mathbf{Q}^{(k)}), D(\mathbf{p}, \mathbf{Q}^{(k)}))$ is arbitrarily close to the segment of the R(D) curve at which the slope is equal to s when k is sufficiently large.