Previously...

- Forward and converse proof of the rate-distortion theorem

This time

- Method of types
- Universal source coding
- Large deviation theory

Project presentation

- Start as usual class time (12/12)
- Please prepare ∼30 minutes presentation. Explain your problem statement. Focus on your approach and result
    - Take a format similar to a conference presentation
- Expect ∼5 minutes Q/A
- Grading
    - Presentation: clarity, structure, references, etc. (10/40)
    - Technical: correctness, depth, novelty, etc. (15/40)
    - Evaluation and results: sound evaluation metric, thoroughness in analysis and experimentation (if any), results and performance (15/40)
- Expectation
    - National conference quality (4/4), reserach day quality (3/4), research meeting quality (2/4), just show up (1/4)

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have neglible probability

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have neglible probability
- However, sometimes we are interested in the probability of getting say 400 heads, even though we know that the probability is neglible

Motivation

- In previous lectures, we have introduced LLN and typical sequences. In a sense that every sequences drawn from a discrete memoryless source is typical
- Take coin tossing as example again, if $Pr(Head) = 0.6$, and we throw the coin 1000 times. We expect that almost all drawn sequences with have about 600 heads. And the rest have neglible probability
- However, sometimes we are interested in the probability of getting say 400 heads, even though we know that the probability is neglible $\rightarrow$ method of types

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values
- For whatever reason, he is not happy until the sum is at least 40,000. If not, he will just throw the dice again for 10,000

Motivation

By the end of the class, we will be able to solve the following nontrivial puzzle

- Tom throws a unbiased dice for 10,000 times and adds all values
- For whatever reason, he is not happy until the sum is at least 40,000. If not, he will just throw the dice again for 10,000
- Now, by the time he eventually got a sequence with sum at least 40,000, *approximately how many ones in the sequence?*

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is
$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$
- How about the probability of getting a particular sequence with 400 heads? It is
$$0.6^{400}0.4^{600} = 2^{-1000(-0.4\log 0.6 - 0.6\log 0.4)}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$

- How about the probability of getting a particular sequence with 400 heads? It is

$$\begin{aligned} 0.6^{400}0.4^{600} &= 2^{-1000(-0.4\log 0.6 - 0.6\log 0.4)} \\ &= 2^{-1000(-0.4\log 0.4 - 0.6\log 0.6 + 0.4\log\frac{0.4}{0.6} + 0.6\log\frac{0.6}{0.4})} \end{aligned}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$

- How about the probability of getting a particular sequence with 400 heads? It is

$$\begin{aligned} 0.6^{400}0.4^{600} &= 2^{-1000(-0.4\log 0.6 - 0.6\log 0.4)} \\ &= 2^{-1000(-0.4\log 0.4 - 0.6\log 0.6 + 0.4\log\frac{0.4}{0.6} + 0.6\log\frac{0.6}{0.4})} \\ &= 2^{-N(H(X) + KL((0.4,0.6)\|(0.6,0.4))} \end{aligned}$$

Type class

Continue with the coin-tossing example

- Recall that the probability of getting a particular sequence with 600 heads is

$$0.6^{600}0.4^{400} = 2^{-1000(-0.6\log 0.6 - 0.4\log 0.4)} = 2^{-NH(X)}$$

- How about the probability of getting a particular sequence with 400 heads? It is

$$\begin{aligned} 0.6^{400}0.4^{600} &= 2^{-1000(-0.4\log 0.6 - 0.6\log 0.4)} \\ &= 2^{-1000(-0.4\log 0.4 - 0.6\log 0.6 + 0.4\log\frac{0.4}{0.6} + 0.6\log\frac{0.6}{0.4})} \\ &= 2^{-N(H(X)+KL((0.4,0.6)\|(0.6,0.4))} \end{aligned}$$

- Every sequence with 400 heads has the same probability. And in general, sequences with the same fraction of outcomes have same probability and we can put them into the same (type) class

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a)$, $\forall a \in \mathcal{X}$

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a)$, $\forall a \in \mathcal{X}$
- Let us reserve $q(x)$ as the true distribution of x (i.e., $q(Head) = 0.6$ and $q(Tail) = 0.4$). And in general, we expect all sequences drawn from the source should belongs to $T(q)$ asymptotically

Type class

- For convenience, let us denote the number of a in the sequence x^N as $\mathscr{N}(a|x^N)$
- Then for any valid distribution of X, $p(x)$, we will define a type class $T(p_X)$ as the set containing all sequences such that $\frac{\mathscr{N}(a|x^N)}{N} \approx p(a)$, $\forall a \in \mathcal{X}$
- Let us reserve $q(x)$ as the true distribution of x (i.e., $q(Head) = 0.6$ and $q(Tail) = 0.4$). And in general, we expect all sequences drawn from the source should belongs to $T(q)$ asymptotically
- Let's also refer p_{x^N} as the empirical distribution of x^N. That is $p_{x^N}(a) = \frac{\mathscr{N}(a|x^N)}{N}$. So $T(p_{x^N})$ is the type class containing x^N

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$,

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3

Example

Let $\mathcal{X} \in \{1, 2, 3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20.$

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$$

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$$

Actually we don't care too much what $|T(p)|$ is exactly. We will provide bounds for $|T(p)|$ as we come back later on

Example

Let $\mathcal{X} \in \{1,2,3\}$ and $x^N = 11321$

- $p_{x^N}(1) = \frac{3}{5}$, $p_{x^N}(2) = \frac{1}{5}$, $p_{x^N}(3) = \frac{1}{5}$
- $T(p_{x^N}) = \{11123, 11132, 11231, 11321, \cdots\}$ containing all sequences with three 1's, one 2, and one 3
- $|T(p_{x^N})| = \frac{5!}{3!1!1!} = 20$. In general,

$$|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$$

Actually we don't care too much what $|T(p)|$ is exactly. We will provide bounds for $|T(p)|$ as we come back later on

- And for any sequence $\mathbf{y}$ in $T(p_{x^N})$, $p(\mathbf{y}) = q(1)^3 q(2) q(3)$, where $q(\cdot)$ is the true distribution

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a \in \mathcal{X}} \mathcal{N}(a|x^N) \log q(a)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N) \log q(a)}$$

$$= 2^{-N \sum_{a\in\mathcal{X}} -p_{x_N}(a) \log q(a)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N) \log q(a)}$$

$$= 2^{-N \sum_{a\in\mathcal{X}} -p_{x_N}(a) \log q(a)} = 2^{-N\left(-\sum_{a\in\mathcal{X}} p(a) \log p(a) - \sum_{a\in\mathcal{X}} p(a) \log \frac{p(a)}{q(a)}\right)}$$

Type sequence probability

Even though we have seen that in the coin toss example, let's restate it more formally.

Theorem 1

If $x^N \in T(p)$ and $q(\cdot)$ is the true distribution of X, the probability of getting x^N from sampling $q(\cdot)$ for N times, as denoted as $q^N(x^N)$, is given by

$$2^{-N(H(p)+KL(p||q))}$$

Proof

$$q^N(x^N) = \prod_{i=1}^{N} q(x_i) = 2^{\sum_{i=1}^{N} \log q(x_i)} = 2^{\sum_{a\in\mathcal{X}} \mathcal{N}(a|x^N)\log q(a)}$$

$$= 2^{-N\sum_{a\in\mathcal{X}} -p_{x_N}(a)\log q(a)} = 2^{-N\left(-\sum_{a\in\mathcal{X}} p(a)\log p(a) - \sum_{a\in\mathcal{X}} p(a)\log\frac{p(a)}{q(a)}\right)}$$

$$= 2^{-N(H(p)+KL(p||q))}$$

Probability of a sequence in the "typical" class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Probability of a sequence in the "typical" class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Remarks

- Note that the probability is exactly equal to $2^{-NH(X)}$

Probability of a sequence in the "typical" class

If $x^N \in T(q)$, where $q(\cdot)$ is the true distribution of X, then

$$q^N(x^N) = 2^{-NH(q)} = 2^{-NH(X)}$$

Remarks

- Note that the probability is exactly equal to $2^{-NH(X)}$
- Recall that this is the probability of a typical sequence supposed to be. Therefore, any x^N in $T(q)$ is a typical sequence ($T(q) \subset A_\epsilon^N(X)$)

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0,1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0, 1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N + 1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0,1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N+1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence
- Each element p of $\mathcal{P}_N(X)$ corresponds a type $T(p)$

Set of all empirical distribution $\mathcal{P}_N(X)$

Denote $\mathcal{P}_N(X)$ as the set of all empirical distribution of X in a length-N sequence

Example

If $X \in \{0,1\}$,

$$\mathcal{P}_N(X) = \left\{ (p_X(0), p_X(1)) : \left(\frac{0}{N}, \frac{N}{N}\right), \left(\frac{1}{N}, \frac{N-1}{N}\right), \cdots, \left(\frac{N}{N}, \frac{0}{N}\right) \right\}$$

Note that $|\mathcal{P}_N(X)| = N+1$

- Since a type is uniquely characterized by a distribution of X in a length-N sequence
- Each element p of $\mathcal{P}_N(X)$ corresponds a type $T(p)$
- Number of types is $|\mathcal{P}_N(X)|$

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively "small"

Theorem 2

$$|\mathcal{P}_N(X)| \le (N+1)^{|\mathcal{X}|}$$

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively "small"

Theorem 2

$$|\mathcal{P}_N(X)| \leq (N+1)^{|\mathcal{X}|}$$

Proof

Note that each type is specified by the empirical probability of each outcome of X. And the possible values of the empirical probabilities are $\frac{0}{N}, \frac{1}{N}, \cdots, \frac{N}{N}$ ($N+1$ of them).

Number of types

It is not too difficult to count the exact number of types. But in practice, we don't quite bother with it as long as we know that the number is relatively "small"

Theorem 2

$$|\mathcal{P}_N(X)| \leq (N+1)^{|\mathcal{X}|}$$

Proof

Note that each type is specified by the empirical probability of each outcome of X. And the possible values of the empirical probabilities are $\frac{0}{N}, \frac{1}{N}, \cdots, \frac{N}{N}$ ($N+1$ of them). Since there are $|\mathcal{X}|$ elements, the number of types is bounded by $(N+1)^{|\mathcal{X}|}$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N)$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p}))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p}))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \le |T(p)| \le 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \ge \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \le \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p)) \leq (N+1)^{|\mathcal{X}|} Pr(T(p))$$

Size of a type class

Recall that $|T(p)| = \frac{N!}{(Np(x_1))!(Np(x_2))!(Np(x_3))!\cdots}$ but the following bounds are much more useful in practice

Theorem 3

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Proof

Let's assume $p(\cdot)$ is the actual distribution of X here

$$1 \geq \sum_{x^N \in T(p)} p^N(x^N) = \sum_{x^N \in T(p)} 2^{-NH(p)} = |T(p)| 2^{-NH(p)}$$

$$1 = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(\hat{p})) \leq \sum_{\hat{p} \in \mathcal{P}_N} \max_{\tilde{p}} Pr(T(\tilde{p})) = \sum_{\hat{p} \in \mathcal{P}_N} Pr(T(p)) \leq (N+1)^{|\mathcal{X}|} Pr(T(p))$$

$$= (N+1)^{|\mathcal{X}|} |T(p)| 2^{-NH(p)}$$

Probability of a type class

Theorem 4

Let the true distribution of X is $q(\cdot)$, then

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Probability of a type class

Theorem 4

Let the true distribution of X is $q(\cdot)$, then

$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Proof

From Theorem 1, each sequence in $T(p)$ has probability $2^{-N(H(p)+KL(p||q))}$ and since $\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$ from Theorem 3,

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} 2^{-N(H(p)+KL(p||q))} \leq Pr(T(p)) \leq 2^{NH(p)} 2^{-N(H(p)+KL(p||q))}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,

$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$

- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)

$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$

- There are about $2^{NH(p)}$ sequences in $T(p)$

$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,
$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$
- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)
$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$
- There are about $2^{NH(p)}$ sequences in $T(p)$
$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$
- Probability of getting a sequence in $T(p)$ is about $2^{-N(KL(p||q))}$. More precisely,
$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$

Summary of type

- Type class $T(p)$ contains all sequences with empirical distribution of p. That is,
$$T(p) = \left\{ x^N : \frac{\mathcal{N}(a|x^N)}{N} = p(a) \right\}$$
- All sequences in the type class $T(p)$ has the same probability ($q(\cdot)$ is the true distribution)
$$q^N(x^N) = 2^{-N(H(p)+KL(p||q)}$$
- There are about $2^{NH(p)}$ sequences in $T(p)$
$$\frac{1}{(N+1)^{|\mathcal{X}|}} 2^{NH(p)} \leq |T(p)| \leq 2^{NH(p)}$$
- Probability of getting a sequence in $T(p)$ is about $2^{-N(KL(p||q))}$. More precisely,
$$\frac{2^{-N(KL(p||q))}}{(N+1)^{|\mathcal{X}|}} \leq Pr(T(p)) \leq 2^{-N(KL(p||q))}$$
- There are $(N+1)^{|\mathcal{X}|}$ types

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder
- Question: Is it possible to construct compression scheme without knowing the source distibution and still performs as good?

Rationale

- For the compression scheme (such as Huffmann coding) that we discussed earlier in this class, one needs to know the source distribution ahead to design the encoder and decoder
- Question: Is it possible to construct compression scheme without knowing the source distibution and still performs as good?
- Answer: Yes. At least theoretically $\rightarrow$ universal source coding

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book.

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)|$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

$$\leq (N+1)^{|\mathcal{X}|} 2^{NR_N}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

$$\leq (N+1)^{|\mathcal{X}|} 2^{NR_N} = 2^{N\left(R_N + |\mathcal{X}|\frac{\log(N+1)}{N}\right)} = 2^{NR}$$

Theory of universal source coding

Given any source Q with $H(Q) < R$, there exists a length-N universal code of rate R such that the source can be decoded losslessly as $N \to \infty$

Proof

Let $R_N = R - |\mathcal{X}|\frac{\log(N+1)}{N}$, and consider the set of sequences $A = \{x^N : H(p_{x^N}) < R_N\}$ as the code book. Note that the rate is $< R$ as

$$|A| = \sum_{p:H(p)<R_N} |T(p)| \leq \sum_{p:H(p)<R_N} 2^{NH(p)} < \sum_{p:H(p)<R_N} 2^{NR_N}$$

$$\leq (N+1)^{|\mathcal{X}|} 2^{NR_N} = 2^{N\left(R_N + |\mathcal{X}|\frac{\log(N+1)}{N}\right)} = 2^{NR}$$

- Encoder: given input, check if input is in A, output index if so. Otherwise, declare failure
- Decoder: simply map index back to the sequence

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p))$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$
$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$
$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$

$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q
- $\Rightarrow \min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q) > 0$ for $N \geq N_0$

Theory of universal source coding

Proof (con't)

Note that the probability of error P_e is given by

$$P_e = \sum_{p:H(p)>R_N} Pr(T(p)) \leq \sum_{p:H(p)>R_N} \max_{\tilde{p}:H(\tilde{p})>R_N} Pr(T(\tilde{p}))$$
$$\leq (1+N)^{|\mathcal{X}|} 2^{-N\left(\min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q)\right)}$$

- If $H(q) < R$, as $R_N \to R$ as N increases, we can find some N_0 such that $H(q) < R_N$ for all $N \geq N_0$
- Therefore, any p in $\{p : H(p) > R_N\}$ cannot be the same as q
- $\Rightarrow \min_{\tilde{p}:H(\tilde{p})>R_N} KL(\tilde{p}||q) > 0$ for $N \geq N_0$
- Hence, $P_e \to 0$ as $N \to \infty$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
    - Construct a dictionary including all previously seen segments

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
  - First parse segment into segments that haven't seen before $\Rightarrow$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
  - First parse segment into segments that haven't seen before ⇒
    $\overset{1}{1}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 10110111011110111
  - First parse segment into segments that haven't seen before $\Rightarrow$ $\overset{1}{1}, \overset{2}{0}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 10110111011110111
  - First parse segment into segments that haven't seen before $\Rightarrow$ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers:

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary;

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101110111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before $\Rightarrow$ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit $\Rightarrow (0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$

Lempel-Ziv coding

- Its variants are widely used by compression tools almost everywhere (zip, pkzip, tiff, etc.)
- Main ideas
  - Construct a dictionary including all previously seen segments
  - Bits needed to send a new segment can be reduced taking advantage known segment in the dictionary
- Example: let's compress 1011011101111 0111
  - First parse segment into segments that haven't seen before ⇒ $\overset{1}{1}, \overset{2}{0}, \overset{3}{11}, \overset{4}{01}, \overset{5}{110}, \overset{6}{111}, \overset{7}{10}, \overset{8}{111}$
  - Encode each segment into representation containing a pair of numbers: 1) index of segment (excluding the last bit) in the dictionary; 2) the last bit ⇒ $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
  - Encode representation to bit stream. Note that as the dictionary grows, number of bits needed to store the index increases ⇒ 0100011101011001110010110

Lempel-Ziv decoding

- Decode bitstream back to representation
  0100011101011001110010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

Lempel-Ziv decoding

- Decode bitstream back to representation
  01000111010110011100010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

$$1$$
$$1$$

$\Rightarrow 1$

Lempel-Ziv decoding

- Decode bitstream back to representation
  01000111010110011100010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2
1	0

$\Rightarrow$ 10

Lempel-Ziv decoding

- Decode bitstream back to representation
  $0100011101011001110010110 \Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3
1	0	11

$\Rightarrow 1011$

Lempel-Ziv decoding

- Decode bitstream back to representation
  0100011101011001110010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4
1	0	11	01

$\Rightarrow$ 101101

Lempel-Ziv decoding

- Decode bitstream back to representation
  010001110101100111001011 0 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5
1	0	11	01	110

$\Rightarrow$ 101101110

Lempel-Ziv decoding

- Decode bitstream back to representation
  010001110101100111001 0110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6
1	0	11	01	110	111

$\Rightarrow$ 101101110111

Lempel-Ziv decoding

- Decode bitstream back to representation
  01000111010110011100010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6	7
1	0	11	01	110	111	10

$\Rightarrow$ 10110111011110

Lempel-Ziv decoding

- Decode bitstream back to representation
  0100011101011001110010110 $\Rightarrow$
  $(0,1),(0,0),(1,1),(2,1),(3,0),(3,1),(1,0),(6,\varnothing)$
- Build dictionary and decode

1	2	3	4	5	6	7	8
1	0	11	01	110	111	10	111

$\Rightarrow$ 10110111011110111

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability
- However, if we insist finding the probability of getting 400 heads, from discussion up to now, we know that it is just

$$Pr(T((0.4,0.6))) \approx 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Motivation

- Let's revisit some coin tossing example. Say if a coin is fair, and we toss if for 1000 times, we know that we will almost always get 500 heads. So getting, say, 400 heads has neglible probability
- However, if we insist finding the probability of getting 400 heads, from discussion up to now, we know that it is just

$$Pr(T((0.4, 0.6))) \approx 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

- Now, what if we are interested in the probability of a more general case? Say what is the probability of getting $>$ 300 and $<$ 400 heads?

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000})$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p))$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$

$$\leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$

$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$

$$\leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Sanov's Theorem

Let $X_1, X_2, \cdots, X_N$ be i.i.d. $\sim q(\cdot)$ and $\mathcal{E}$ be a set of distribution. Then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_N) \leq (N+1)^{|\mathcal{X}|} 2^{-N(KL(p^*||q))},$$

where $p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$.

Sanov's Theorem

Let $\mathcal{E} = \{p : 0.3 \leq p(Head) \leq 0.4\}$ and $q(\cdot) = (0.5, 0.5)$ is the true distribution, then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_{1000}) = \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} Pr(T(p)) \approx \sum_{p \in \mathcal{E} \cap \mathcal{P}_{1000}} 2^{-1000(KL(p||q))}$$
$$= 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))} + 2^{-1000(KL((0.399,0.601)||(0.5,0.5)))} + 2^{-1000(KL((0.398,0.602)||(0.5,0.5)))} + \cdots + 2^{-1000(KL((0.3,0.7)||(0.5,0.5)))}$$
$$\leq |\mathcal{P}_{1000}| 2^{-1000(KL((0.4,0.6)||(0.5,0.5)))}$$

Sanov's Theorem

Let $X_1, X_2, \cdots, X_N$ be i.i.d. $\sim q(\cdot)$ and $\mathcal{E}$ be a set of distribution. Then

$$Pr(\mathcal{E}) = Pr(\mathcal{E} \cap \mathcal{P}_N) \leq (N+1)^{|\mathcal{X}|} 2^{-N(KL(p^*||q))},$$

where $p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$. Moreover, given a rather weak condition (closure of interior of $\mathcal{E}$ is $\mathcal{E}$ itself), we have

$$\frac{1}{N} \log Pr(\mathcal{E}) \to -KL(p^*||q)$$

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

Let $\mathcal{E}$ be a closed convex subset of $\mathcal{P}$ (the set of all distributions) and $q(\cdot)$ be the true distribution which is $\notin \mathcal{E}$.

Conditional limit theorem

- The first part of Sanov's Theorm is easy to show as similar to the example. However, the second half will need some more math background (mostly mathematical analysis) to understand the proof and so we will skip it here
- The latter part of Sanov's Theorem suggests that the probability of getting $\mathcal{E}$ is the same as the probability of getting $T(p^*)$
- It turns out that we can claim something stronger. We will state the theorem below without proof

Conditional limit theorem

Let $\mathcal{E}$ be a closed convex subset of $\mathcal{P}$ (the set of all distributions) and $q(\cdot)$ be the true distribution which is $\notin \mathcal{E}$. If $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$ and we know that $p_{x_N} \in \mathcal{E}$, then

$$\frac{\mathcal{N}(a|x_N)}{N} \to p^*(a)$$

in probability as $N \to \infty$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall

$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \le p(Head) \le 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p\in\mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$
- By conditional limit theorem, knowing the the number of head is within the range, the coin behaves like a biased coin with $p(Head) = 0.4$

Examples

Coin toss

- Let's go back to our previous example. If we throw a fair coin 1000 times and some one tells you that there are 300 to 400 heads, recall
$$\mathcal{E} = \{0.3 \leq p(Head) \leq 0.4\}$$
- Since apparently,
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||(0.5, 0.5)) = (0.4, 0.6)$$
- By conditional limit theorem, knowing the the number of head is within the range, the coin behaves like a biased coin with $p(Head) = 0.4$
- A best bet would be there are 400 heads

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\frac{1}{N}\sum_{i=1}^{N} g_k(x_i) \geq \alpha_k$$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\frac{1}{N}\sum_{i=1}^{N} g_k(x_i) \ge \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a) g_k(a) \ge \alpha_k, k = 1, \cdots, K\}$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\frac{1}{N}\sum_{i=1}^{N} g_k(x_i) \geq \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a) g_k(a) \geq \alpha_k, k = 1, \cdots, K\}$
- From conditional limit theorem, $\frac{\mathcal{N}(a|x^N)}{N} \to p^*(a)$, where
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$$

Examples

Lower bounds

- Let say $x_1, x_2, \cdots, x_N$ are drawn from $q(\cdot)$. And we have K functions $g_1(\cdot), g_2(\cdot), \cdots, g_K(\cdot)$ such that for $k = 1, \cdots, K$,
$$\frac{1}{N}\sum_{i=1}^{N} g_k(x_i) \geq \alpha_k$$
- Let $\mathcal{E} = \{p : \sum_a p(a) g_k(a) \geq \alpha_k, k = 1, \cdots, K\}$
- From conditional limit theorem, $\frac{\mathcal{N}(a|x^N)}{N} \to p^*(a)$, where
$$p^* = \arg\min_{p \in \mathcal{E}} KL(p||q)$$
- This is a simple constrained optimization problem and can be solved with KKT conditions. If you go through the conditions, you will find that
$$p^*(x) \propto q(x) 2^{\sum_{k=1}^{K} \lambda_k g_k(x)},$$
with $\lambda_k (\sum_a p(a) g_k(a) - \alpha_k) = 0$, $\lambda_k \geq 0$, and $\sum_a p(a) g_k(a) \geq \alpha_k$

Examples

I think this example below gives a nice demonstration that the technique we have learned today can solve some amazing puzzle!

Examples

I think this example below gives a nice demonstration that the technique we have learned today can solve some amazing puzzle!

Fair dice

A fair dice is thrown 10,000 times and the sum of all outcomes is larger than 40,000, out of the 10,000 throw, how many ones do you think there are?

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

for some λ

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

for some λ

- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^6 2^{\lambda j}}$$

for some λ

- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k(\sum_a p(a) g_k(a) - \alpha_k) = 0$,

$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect
$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$
for some λ
- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k (\sum_a p(a) g_k(a) - \alpha_k) = 0$,
$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$
- This gives us $\lambda = 0.2519$, and thus $p^* = (0.103, 0.123, 0.146, 0.174, 0.207, 0.247)$

Fair dice

- From the result of previous example, let $g_1(x) = x$ and $\alpha_1 = 4$, we expect

$$p^*(i) = \frac{2^{\lambda i}}{\sum_{j=1}^{6} 2^{\lambda j}}$$

  for some λ

- $\lambda \neq 0$ since $\sum_a p(a) g_1(a) = 3.5 < 4 = \alpha_1$ if so
- Since $\lambda \neq 0$, by the complementary slackness constraint $\lambda_k (\sum_a p(a) g_k(a) - \alpha_k) = 0$,

$$\sum_a p(a) g_1(a) = \alpha_1 = 4$$

- This gives us $\lambda = 0.2519$, and thus $p^* = (0.103, 0.123, 0.146, 0.174, 0.207, 0.247)$
- # ones $\approx 0.103 \times 10000 = 1030$