Entropy, Randomness, and Information

Lecture 27
December 5, 2013
Part I

Entropy
“If only once - only once - no matter where, no matter before what audience - I could better the record of the great Rastelli and juggle with thirteen balls, instead of my usual twelve, I would feel that I had truly accomplished something for my country. But I am not getting any younger, and although I am still at the peak of my powers there are moments - why deny it? - when I begin to doubt - and there is a time limit on all of us.”
The entropy in bits of a discrete random variable \( X \) is

\[
\mathbb{H}(X) = - \sum_x \Pr[X = x] \log \Pr[X = x].
\]

Equivalently,

\[
\mathbb{H}(X) = \mathbb{E} \left[ \log \frac{1}{\Pr[X]} \right].
\]
Entropy intuition...

Intuition...

$H(X)$ is the number of fair coin flips that one gets when getting the value of $X$. 
Binary entropy

\( H(X) = - \sum_x \Pr[X = x] \lg \Pr[X = x] \)

Definition

The **binary entropy** function \( H(p) \) for a random binary variable that is 1 with probability \( p \), is \( H(p) = -p \lg p - (1 - p) \lg(1 - p) \). We define \( H(0) = H(1) = 0 \).

Q: How many truly random bits are there when given the result of flipping a single coin with probability \( p \) for heads?
Binary entropy

$$H(X) = - \sum_x \Pr[X = x] \lg \Pr[X = x]$$

Definition

The **binary entropy** function $H(p)$ for a random binary variable that is 1 with probability $p$, is $H(p) = -p \lg p - (1 - p) \lg (1 - p)$. We define $H(0) = H(1) = 0$.

Q: How many truly random bits are there when given the result of flipping a single coin with probability $p$ for heads?
Binary entropy

\[ H(X) = - \sum_x \Pr[X = x] \log \Pr[X = x] \]

Definition

The **binary entropy** function \( H(p) \) for a random binary variable that is \( 1 \) with probability \( p \), is \( H(p) = -p \log p - (1 - p) \log(1 - p) \). We define \( H(0) = H(1) = 0 \).

Q: How many truly random bits are there when given the result of flipping a single coin with probability \( p \) for heads?
Binary entropy:
\[ H(p) = -p \log p - (1 - p) \log (1 - p) \]

\( H(p) \) is a concave symmetric around \( 1/2 \) on the interval \([0, 1]\).

- Maximum at \( 1/2 \).
- \( H(3/4) \approx 0.8113 \) and \( H(7/8) \approx 0.5436 \).
- A coin that has \( 3/4 \) probably to be heads have higher amount of “randomness” in it than a coin that has probability \( 7/8 \) for heads.
Binary entropy:

$$\mathcal{H}(p) = -p \log_2 p - (1 - p) \log_2 (1 - p)$$

1. $\mathcal{H}(p)$ is a concave symmetric around $1/2$ on the interval $[0, 1]$.
2. Maximum at $1/2$.
3. $\mathcal{H}(3/4) \approx 0.8113$ and $\mathcal{H}(7/8) \approx 0.5436$.
4. Coin that has $3/4$ probably to be heads have higher amount of “randomness” in it than a coin that has probability $7/8$ for heads.
Binary entropy:
\[ H(p) = -p \log p - (1 - p) \log(1 - p) \]

1. \( H(p) \) is a concave symmetric around \( 1/2 \) on the interval \([0, 1]\).
2. maximum at \( 1/2 \).
3. \( H(3/4) \approx 0.8113 \) and \( H(7/8) \approx 0.5436 \).
4. coin that has \( 3/4 \) probably to be heads have higher amount of “randomness” in it than a coin that has probability \( 7/8 \) for heads.
Binary entropy:

$$H(p) = -p \log p - (1 - p) \log(1 - p)$$

1. $H(p)$ is a concave symmetric around $1/2$ on the interval $[0, 1]$.
2. maximum at $1/2$.
3. $H(3/4) \approx 0.8113$ and $H(7/8) \approx 0.5436$.
4. coin that has $3/4$ probably to be heads have higher amount of “randomness” in it than a coin that has probability $7/8$ for heads.
And now for some unnecessary math

1. $H(p) = -p \log p - (1 - p) \log(1 - p)$
2. $H'(p) = -\log p + \log(1 - p) = \log \frac{1-p}{p}$
3. $H''(p) = \frac{p}{1-p} \cdot \left( -\frac{1}{p^2} \right) = -\frac{1}{p(1-p)}$.
4. $\implies H''(p) \leq 0$, for all $p \in (0, 1)$, and the $H(\cdot)$ is concave.
5. $H'(1/2) = 0 \implies H(1/2) = 1$ max of binary entropy.
6. $\implies$ balanced coin has the largest amount of randomness in it.
And now for some unnecessary math

1. \( H(p) = -p \log p - (1 - p) \log(1 - p) \)
2. \( H'(p) = -\log p + \log(1 - p) = \log \frac{1-p}{p} \)
3. \( H''(p) = \frac{p}{1-p} \cdot \left(-\frac{1}{p^2}\right) = -\frac{1}{p(1-p)} \).
4. \( \implies H''(p) \leq 0 \), for all \( p \in (0, 1) \), and the \( H(\cdot) \) is concave.
5. \( H'(1/2) = 0 \implies H(1/2) = 1 \) max of binary entropy.
6. \( \implies \) balanced coin has the largest amount of randomness in it.
And now for some unnecessary math

1. \( H(p) = -p \log p - (1 - p) \log (1 - p) \)

2. \( H'(p) = - \log p + \log (1 - p) = \log \frac{1-p}{p} \)

3. \( H''(p) = \frac{p}{1-p} \cdot \left( -\frac{1}{p^2} \right) = -\frac{1}{p(1-p)} \).

\( \implies H''(p) \leq 0, \) for all \( p \in (0, 1) \), and the \( H(\cdot) \) is concave.

4. \( H'(1/2) = 0 \implies H(1/2) = 1 \) max of binary entropy.

5. \( H'(1/2) = 0 \implies H(1/2) = 1 \) max of binary entropy.

6. \( \implies \) balanced coin has the largest amount of randomness in it.
And now for some unnecessary math

1. $H(p) = -p \log p - (1 - p) \log (1 - p)$
2. $H'(p) = -\log p + \log (1 - p) = \log \frac{1 - p}{p}$
3. $H''(p) = \frac{p}{1 - p} \cdot \left(-\frac{1}{p^2}\right) = -\frac{1}{p(1-p)}.$
4. $\implies H''(p) \leq 0$, for all $p \in (0,1)$, and the $H(\cdot)$ is concave.
5. $H'(1/2) = 0 \implies H(1/2) = 1$ max of binary entropy.
6. $\implies$ balanced coin has the largest amount of randomness in it.
And now for some unnecessary math

1. $H(p) = -p \log p - (1 - p) \log(1 - p)$
2. $H'(p) = -\log p + \log(1 - p) = \log \frac{1-p}{p}$
3. $H''(p) = \frac{1}{1-p} \cdot \left( -\frac{1}{p^2} \right) = -\frac{1}{p(1-p)}$.
4. $\implies H''(p) \leq 0$, for all $p \in (0, 1)$, and the $H(\cdot)$ is concave.
5. $H'(1/2) = 0 \implies H(1/2) = 1$ max of binary entropy.
6. $\implies$ balanced coin has the largest amount of randomness in it.
And now for some unnecessary math

1. \( H(p) = -p \log p - (1 - p) \log (1 - p) \)

2. \( H'(p) = -\log p + \log (1 - p) = \log \frac{1-p}{p} \)

3. \( H''(p) = \frac{p}{1-p} \cdot \left( -\frac{1}{p^2} \right) = -\frac{1}{p(1-p)} \)

4. \( \implies H''(p) \leq 0 \), for all \( p \in (0, 1) \), and the \( H(\cdot) \) is concave.

5. \( H'(1/2) = 0 \implies H(1/2) = 1 \text{ max} \) of binary entropy.

6. \( \implies \text{balanced coin has the largest amount of randomness in it.} \)
Squeezing good random bits out of bad random bits...

Given the result of $n$ coin flips: $b_1, \ldots, b_n$ from a faulty coin, with head with probability $p$, how many truly random bits can we extract?
Squeezing good random bits out of bad random bits...

**Question…**

Given the result of \( n \) coin flips: \( b_1, \ldots, b_n \) from a faulty coin, with head with probability \( p \), how many truly random bits can we extract?

If believe intuition about entropy, then this number should be

\[
\approx n H(p).
\]
entropy of $X$ is $H(X) = - \sum_x \Pr[X = x] \log \Pr[X = x]$. 

Example

A random variable $X$ that has probability $1/n$ to be $i$, for $i = 1, \ldots, n$, has entropy $H(X) = - \sum_{i=1}^n \frac{1}{n} \log \frac{1}{n} = \log n$. 

Entropy is oblivious to the exact values random variable can have.

random variables over $-1, +1$ with equal probability has the same entropy (i.e., 1) as a fair coin.
1. **Entropy** of $X$ is $H(X) = -\sum_x \Pr[X = x] \log \Pr[X = x]$.

2. Entropy of uniform variable.

**Example**

A random variable $X$ that has probability $\frac{1}{n}$ to be $i$, for $i = 1, \ldots, n$, has entropy $H(X) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = \log n$.

3. Entropy is oblivious to the exact values random variable can have.

4. Random variables over $-1, +1$ with equal probability has the same entropy (i.e., 1) as a fair coin.
entropy of $X$ is $\mathbb{H}(X) = -\sum_x \Pr[X = x] \log \Pr[X = x]$.

Entropy of uniform variable.

Example

A random variable $X$ that has probability $1/n$ to be $i$, for $i = 1, \ldots, n$, has entropy $\mathbb{H}(X) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = \log n$.

Entropy is oblivious to the exact values random variable can have.

random variables over $-1, +1$ with equal probability has the same entropy (i.e., $1$) as a fair coin.
1. **Entropy** of $X$ is $H(X) = - \sum_x \Pr[X = x] \log \Pr[X = x]$.

2. Entropy of uniform variable.

**Example**

A random variable $X$ that has probability $\frac{1}{n}$ to be $i$, for $i = 1, \ldots, n$, has entropy $H(X) = - \sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = \log n$.

3. Entropy is oblivious to the exact values random variable can have.

4. Random variables over $-1, 1$ with equal probability has the same entropy (i.e., 1) as a fair coin.
1. **Entropy** of $X$ is $H(X) = -\sum_x \Pr[X = x] \log \Pr[X = x]$.

2. Entropy of uniform variable.

### Example

A random variable $X$ that has probability $\frac{1}{n}$ to be $i$, for $i = 1, \ldots, n$, has entropy $H(X) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = \log n$.

3. Entropy is oblivious to the exact values random variable can have.

4. Random variables over $-1, +1$ with equal probability has the same entropy (i.e., $1$) as a fair coin.
Lemma

Let $X$ and $Y$ be two independent random variables, and let $Z$ be the random variable $(X, Y)$. Then $\mathbb{H}(Z) = \mathbb{H}(X) + \mathbb{H}(Y)$. 
Proof

In the following, summation are over all possible values that the variables can have. By the independence of $X$ and $Y$ we have

$$H(Z) = \sum_{x,y} \Pr[(X, Y) = (x, y)] \log \frac{1}{\Pr[(X, Y) = (x, y)]}$$

$$= \sum_{x,y} \Pr[X = x] \Pr[Y = y] \log \frac{1}{\Pr[X = x] \Pr[Y = y]}$$

$$= \sum_x \sum_y \Pr[X = x] \Pr[Y = y] \log \frac{1}{\Pr[X = x]}$$

$$+ \sum_y \sum_x \Pr[X = x] \Pr[Y = y] \log \frac{1}{\Pr[Y = y]}$$
Proof continued

\[ H(Z) = \sum_x \sum_y \Pr[X = x] \Pr[Y = y] \lg \frac{1}{\Pr[X = x]} \]

\[ + \sum_y \sum_x \Pr[X = x] \Pr[Y = y] \lg \frac{1}{\Pr[Y = y]} \]

\[ = \sum_x \Pr[X = x] \lg \frac{1}{\Pr[X = x]} \]

\[ + \sum_y \Pr[Y = y] \lg \frac{1}{\Pr[Y = y]} \]

\[ = H(X) + H(Y). \]
Bounding the binomial coefficient using entropy

Lemma

Suppose that $nq$ is integer in the range $[0, n]$. Then

$$\frac{2^n \mathbb{H}(q)}{n + 1} \leq \binom{n}{nq} \leq 2^n \mathbb{H}(q).$$
Proof

Holds if $q = 0$ or $q = 1$, so assume $0 < q < 1$. We have

$$\binom{n}{nq} q^{nq} (1 - q)^{n-nq} \leq (q + (1 - q))^n = 1.$$ 

As such, since

$$q^{-nq} (1 - q)^{-(1-q)n} = 2^{n(-q \log q - (1-q) \log(1-q))} = 2^{n \mathbb{H}(q)},$$

we have

$$\binom{n}{nq} \leq q^{-nq} (1 - q)^{-(1-q)n} = 2^{n \mathbb{H}(q)}.$$
Proof

Holds if \( q = 0 \) or \( q = 1 \), so assume \( 0 < q < 1 \). We have

\[
\binom{n}{nq} q^{nq} (1 - q)^{n - nq} \leq (q + (1 - q))^n = 1.
\]

As such, since

\[
q^{-nq} (1 - q)^{-(1-q)n} = 2^n (-q \lg q - (1-q) \lg(1-q)) = 2^n H(q),
\]

we have

\[
\binom{n}{nq} \leq q^{-nq} (1 - q)^{-(1-q)n} = 2^n H(q).
\]
Proof

Holds if \( q = 0 \) or \( q = 1 \), so assume \( 0 < q < 1 \). We have

\[
\binom{n}{nq} q^{nq} (1 - q)^{n - nq} \leq (q + (1 - q))^n = 1.
\]

As such, since

\[
q^{-nq} (1 - q)^{-q^n} = 2^{n(-q \lg q - (1-q) \lg (1-q))} = 2^n H(q),
\]

we have

\[
\binom{n}{nq} \leq q^{-nq} (1 - q)^{-q^n} = 2^n H(q).
\]
Proof

Holds if $q = 0$ or $q = 1$, so assume $0 < q < 1$. We have

\[
\binom{n}{nq} q^{nq} (1 - q)^{n-nq} \leq (q + (1 - q))^n = 1.
\]

As such, since

\[
q^{-nq} (1 - q)^{-(1-q)n} = 2^n (-q \log q - (1-q) \log (1-q)) = 2^n H(q),
\]

we have

\[
\binom{n}{nq} \leq q^{-nq} (1 - q)^{-(1-q)n} = 2^n H(q).
\]
Proof continued

Other direction...

1. $\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}$

2. $\sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i)$.

3. Claim: $\mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq}$ largest term in $\sum_{k=0}^{n} \mu(k) = 1$.

4. $\Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k}(1 - \frac{n-k}{k+1} \frac{q}{1-q})$.

5. sign of $\Delta_k = $ size of last term...

6. $\text{sign}(\Delta_k) = \text{sign}(1 - \frac{(n-k)q}{(k+1)(1-q)})$

   $\quad = \text{sign}(\frac{(k+1)(1-q) - (n-k)q}{(k+1)(1-q)})$. 

Sariel (UIUC) CS573 16 Fall 2013 16 / 28
Proof continued

Other direction...

1. \[ \mu(k) = \binom{n}{k} q^k (1 - q)^{n-k} \]

2. \[ \sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i). \]

3. Claim: \( \mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq} \) largest term in \( \sum_{k=0}^{n} \mu(k) = 1. \)

4. \( \Delta_k = \mu(k) - \mu(k+1) = \binom{n}{k} q^k (1 - q)^{n-k} \left(1 - \frac{n-k}{k+1} \frac{q}{1-q}\right), \)

5. sign of \( \Delta_k \) = size of last term...

6. \[ \text{sign}(\Delta_k) = \text{sign} \left(1 - \frac{(n-k)q}{(k+1)(1-q)}\right) \]

\[ = \text{sign} \left(\frac{(k+1)(1-q)-(n-k)q}{(k+1)(1-q)}\right). \]
Proof continued

Other direction...

1. \( \mu(k) = \binom{n}{k} q^k (1 - q)^{n-k} \)

2. \( \sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i) \).

3. Claim: \( \mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq} \) largest term in \( \sum_{k=0}^{n} \mu(k) = 1 \).

4. \( \Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k} \left(1 - \frac{n-k}{k+1} \frac{q}{1-q}\right) \).

5. sign of \( \Delta_k \) = size of last term...

6. \( \text{sign}(\Delta_k) = \text{sign} \left( 1 - \frac{(n-k)q}{(k+1)(1-q)} \right) \)

\[ \quad = \text{sign} \left( \frac{(k+1)(1-q)-(n-k)q}{(k+1)(1-q)} \right). \]
Proof continued

Other direction...

1. \( \mu(k) = \binom{n}{k} q^k (1 - q)^{n-k} \)

2. \( \sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i) \).

3. Claim: \( \mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq} \) largest term in \( \sum_{k=0}^{n} \mu(k) = 1 \).

4. \( \Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k} \left( 1 - \frac{n-k}{k+1} \frac{q}{1-q} \right) \),

5. sign of \( \Delta_k \) = size of last term...

6. \( \text{sign}(\Delta_k) = \text{sign} \left( 1 - \frac{(n-k)q}{(k+1)(1-q)} \right) \)

\[ = \text{sign} \left( \frac{(k+1)(1-q) - (n-k)q}{(k+1)(1-q)} \right) \]
Proof continued

Other direction...

1. \( \mu(k) = \binom{n}{k} q^k (1 - q)^{n-k} \)

2. \[ \sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i). \]

3. Claim: \( \mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq} \) largest term in \( \sum_{k=0}^{n} \mu(k) = 1. \)

4. \( \Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k} \left( 1 - \frac{n-k}{k+1} \frac{q}{1-q} \right), \)

5. sign of \( \Delta_k = \) size of last term...

6. \[ \text{sign}(\Delta_k) = \text{sign} \left( 1 - \frac{(n-k)q}{(k+1)(1-q)} \right) \]

\[ = \text{sign} \left( \frac{(k+1)(1-q)-(n-k)q}{(k+1)(1-q)} \right). \]
Proof continued
Other direction...

1. $$\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}$$

2. $$\sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i).$$

3. Claim: $$\mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq}$$ largest term in $$\sum_{k=0}^{n} \mu(k) = 1.$$

4. $$\Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k} \left(1 - \frac{n-k}{k+1} \frac{q}{1-q}\right),$$

5. sign of $$\Delta_k$$ = size of last term...

6. $$\text{sign}(\Delta_k) = \text{sign}\left(1 - \frac{(n-k)q}{(k+1)(1-q)}\right)$$
   
   $$= \text{sign}\left(\frac{(k+1)(1-q) - (n-k)q}{(k+1)(1-q)}\right).$$
1. \( \mu(k) = \binom{n}{k} q^k (1 - q)^{n-k} \)

2. \( \sum_{i=0}^{n} \binom{n}{i} q^i (1 - q)^{n-i} = \sum_{i=0}^{n} \mu(i) \).

3. Claim: \( \mu(nq) = \binom{n}{nq} q^{nq} (1 - q)^{n-nq} \) largest term in \( \sum_{k=0}^{n} \mu(k) = 1 \).

4. \( \Delta_k = \mu(k) - \mu(k + 1) = \binom{n}{k} q^k (1 - q)^{n-k} \left( 1 - \frac{n-k}{k+1} \frac{q}{1-q} \right) \),

5. sign of \( \Delta_k = \) size of last term...

6. \( \text{sign}(\Delta_k) = \text{sign}\left( 1 - \frac{(n-k)q}{(k+1)(1-q)} \right) \)

   \[= \text{sign}\left( \frac{(k+1)(1-q) - (n-k)q}{(k+1)(1-q)} \right) \].
1. \[(k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\]
2. \[\Rightarrow \Delta_k \geq 0 \text{ when } k \geq nq + q - 1\]
   \[\Delta_k < 0 \text{ otherwise.}\]
3. \[\mu(k) = \binom{n}{k}q^k(1 - q)^{n-k}\]
4. \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]
5. \[\Rightarrow \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]
6. \[\mu(nq) \text{ larger than the average in sum.}\]
7. \[\Rightarrow \binom{n}{k}q^k(1 - q)^{n-k} \geq \frac{1}{n+1}.\]
8. \[\Rightarrow \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq}(1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^{nH(q)}.\]
Proof continued

1. \((k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\)

2. \(\implies \Delta_k \geq 0\) when \(k \geq nq + q - 1\) \(\Delta_k < 0\) otherwise.

3. \(\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\)

4. \(\mu(k) < \mu(k + 1),\) for \(k < nq,\) and \(\mu(k) \geq \mu(k + 1)\) for \(k \geq nq.\)

5. \(\implies \mu(nq)\) is the largest term in \(\sum_{k=0}^{n} \mu(k) = 1.\)

6. \(\mu(nq)\) larger than the average in sum.

7. \(\implies \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\)

8. \(\implies \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq} (1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^{n\mathbb{H}(q)}.\)
Proof continued

1. \[(k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\]

2. \[\Rightarrow \Delta_k \geq 0 \text{ when } k \geq nq + q - 1 \]
   \[\Delta_k < 0 \text{ otherwise.}\]

3. \[\mu(k) = \binom{n}{k}q^k(1 - q)^{n-k}\]

4. \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]

5. \[\Rightarrow \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]

6. \[\mu(nq) \text{ larger than the average in sum.}\]

7. \[\Rightarrow \binom{n}{k}q^k(1 - q)^{n-k} \geq \frac{1}{n+1}.\]

8. \[\Rightarrow \binom{n}{nq} \geq \frac{1}{n+1}q^{-nq}(1 - q)^{-(n-nq)} = \frac{1}{n+1}2^{nH(q)}.\]
Proof continued

1. \((k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\)

2. \[\implies \Delta_k \geq 0 \text{ when } k \geq nq + q - 1\]
   \[\Delta_k < 0 \text{ otherwise.}\]

3. \(\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\)

4. \(\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\)

5. \[\implies \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]

6. \(\mu(nq) \text{ larger than the average in sum.}\)

7. \[\implies \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\]

8. \[\implies \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq} (1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^n H(q).\]
\begin{enumerate}
\item \[(k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\]
\item \[\implies \Delta_k \geq 0 \text{ when } k \geq nq + q - 1 \]
\[\Delta_k < 0 \text{ otherwise.}\]
\item \[\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\]
\item \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]
\item \[\implies \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]
\item \[\mu(nq) \text{ larger than the average in sum.}\]
\item \[\implies \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\]
\item \[\implies \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq} (1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^n H(q).\]
\end{enumerate}
Proof continued

1. \[(k + 1)(1 - q) - (n - k)q =
   \[k + 1 - kq - q - nq + kq = 1 + k - q - nq.
\]

2. \[\iff \quad \Delta_k \geq 0 \text{ when } k \geq nq + q - 1\]
   \[\Delta_k < 0 \text{ otherwise.}\]

3. \[\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\]

4. \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]

5. \[\iff \quad \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]

6. \[\mu(nq) \text{ larger than the average in sum.}\]

7. \[\iff \quad \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\]

8. \[\iff \quad \frac{1}{n+1} q^{-nq} (1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^n H(q).\]
Proof continued

1. \[(k + 1)(1 - q) - (n - k)q =
   k + 1 - kq - q - nq + kq = 1 + k - q - nq.\]
2. \[\Delta_k \geq 0 \text{ when } k \geq nq + q - 1\]
   \[\Delta_k < 0 \text{ otherwise.}\]
3. \[\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\]
4. \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]
5. \[\Rightarrow \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]
6. \[\mu(nq) \text{ larger than the average in sum.}\]
7. \[\Rightarrow \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\]
8. \[\Rightarrow \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq}(1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^{n\mathbb{H}(q)}.\]
(1) \[(k + 1)(1 - q) - (n - k)q = k + 1 - kq - q - nq + kq = 1 + k - q - nq.\]

2 \[\iff \Delta_k \geq 0 \text{ when } k \geq nq + q - 1\]
\[\Delta_k < 0 \text{ otherwise.}\]

3 \[\mu(k) = \binom{n}{k} q^k (1 - q)^{n-k}\]

4 \[\mu(k) < \mu(k + 1), \text{ for } k < nq, \text{ and } \mu(k) \geq \mu(k + 1) \text{ for } k \geq nq.\]

5 \[\implies \mu(nq) \text{ is the largest term in } \sum_{k=0}^{n} \mu(k) = 1.\]

6 \[\mu(nq) \text{ larger than the average in sum.}\]

7 \[\implies \binom{n}{k} q^k (1 - q)^{n-k} \geq \frac{1}{n+1}.\]

8 \[\implies \binom{n}{nq} \geq \frac{1}{n+1} q^{-nq} (1 - q)^{-(n-nq)} = \frac{1}{n+1} 2^{nH(q)}.\]
Corollary

We have:

(i) \( q \in [0, 1/2] \Rightarrow \binom{n}{\lfloor nq \rfloor} \leq 2^n\mathbb{H}(q). \)

(ii) \( q \in [1/2, 1] \Rightarrow \binom{n}{\lceil nq \rceil} \leq 2^n\mathbb{H}(q). \)

(iii) \( q \in [1/2, 1] \Rightarrow \frac{2^{n\mathbb{H}(q)}}{n+1} \leq \binom{n}{\lfloor nq \rfloor}. \)

(iv) \( q \in [0, 1/2] \Rightarrow \frac{2^{n\mathbb{H}(q)}}{n+1} \leq \binom{n}{\lceil nq \rceil}. \)

Proof is straightforward but tedious.
What we have...

1. Proved that $\binom{n}{nq} \approx 2^{n\mathbb{H}(q)}$.
2. Estimate is loose.
3. Sanity check...

   (I) A sequence of $n$ bits generated by coin with probability $q$ for head.

   (II) By Chernoff inequality... roughly $nq$ heads in this sequence.

   (III) Generated sequence $Y$ belongs to $\binom{n}{nq} \approx 2^{n\mathbb{H}(q)}$ possible sequences.

   (IV) ...of similar probability.

   (V) $\implies \mathbb{H}(Y) \approx \log_2 \binom{n}{nq} = n\mathbb{H}(q)$. 

Sariel (UIUC)  
CS573  
Fall 2013
What we have...

1. Proved that \( \binom{n}{nq} \approx 2^{nH(q)} \).

2. Estimate is loose.

3. Sanity check...
   
   (I) A sequence of \( n \) bits generated by coin with probability \( q \) for head.
   
   (II) By Chernoff inequality... roughly \( nq \) heads in this sequence.
   
   (III) Generated sequence \( Y \) belongs to \( \binom{n}{nq} \approx 2^{nH(q)} \) possible sequences.
   
   (IV) ...of similar probability.
   
   (V) \( \implies H(Y) \approx \log \binom{n}{nq} = nH(q) \).
What we have...

1. Proved that \( \binom{n}{nq} \approx 2^{nH(q)} \).
2. Estimate is loose.
3. Sanity check...

   (I) A sequence of \( n \) bits generated by coin with probability \( q \) for head.

   (II) By Chernoff inequality... roughly \( nq \) heads in this sequence.

   (III) Generated sequence \( Y \) belongs to \( \binom{n}{nq} \approx 2^{nH(q)} \) possible sequences.

   (IV) ...of similar probability.

   (V) \( \implies H(Y) \approx \lg \binom{n}{nq} = nH(q) \).
What we have...

1. Proved that $\binom{n}{nq} \approx 2^{nH(q)}$.
2. Estimate is loose.
3. Sanity check...

   (I) A sequence of $n$ bits generated by coin with probability $q$ for head.
   (II) By Chernoff inequality... roughly $nq$ heads in this sequence.
   (III) Generated sequence $Y$ belongs to $\binom{n}{nq} \approx 2^{nH(q)}$ possible sequences.
   (IV) ...of similar probability.

   (V) $\implies H(Y) \approx \lg \binom{n}{nq} = nH(q)$. 
What we have...

1. Proved that \( \binom{n}{nq} \approx 2^{nH(q)} \).

2. Estimate is loose.

3. Sanity check...

   (I) A sequence of \( n \) bits generated by coin with probability \( q \) for head.

   (II) By Chernoff inequality... roughly \( nq \) heads in this sequence.

   (III) Generated sequence \( Y \) belongs to \( \binom{n}{nq} \approx 2^{nH(q)} \) possible sequences .

   (IV) ...of similar probability.

   (V) \( \implies H(Y) \approx \lg \binom{n}{nq} = nH(q) \).
Extracting randomness...

Entropy can be interpreted as the amount of unbiased random coin flips can be extracted from a random variable.

**Definition**

An extraction function $\text{Ext}$ takes as input the value of a random variable $X$ and outputs a sequence of bits $y$, such that

$$\Pr[\text{Ext}(X) = y \mid |y| = k] = \frac{1}{2^k},$$

whenever $\Pr[|y| = k] > 0$, where $|y|$ denotes the length of $y$. 
Extracting randomness...

1. $X$: uniform random integer variable out of $0, \ldots, 7$.
2. $\text{Ext}(X)$: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. $X$: uniform random integer variable $0, \ldots, 11$.
5. $\text{Ext}(x)$: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between $8$ and $11$?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...

$$\Pr[\text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2] = \frac{1}{4},$$
Extracting randomness...

1. $X$: uniform random integer variable out of $0, \ldots, 7$.
2. $\text{Ext}(X)$: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. $X$: uniform random integer variable $0, \ldots, 11$.
5. $\text{Ext}(x)$: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between $8$ and $11$?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...

$$\Pr \left[ \text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2 \right] = \frac{1}{4},$$
Extracting randomness...

1. $X$: uniform random integer variable out of $0, \ldots, 7$.
2. $\text{Ext}(X)$: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. $X$: uniform random integer variable $0, \ldots, 11$.
5. $\text{Ext}(x)$: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between 8 and 11?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...

\[
\Pr\left[ \text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2 \right] = \frac{1}{4},
\]
Extracting randomness...

1. $X$: uniform random integer variable out of $0, \ldots, 7$.
2. $\text{Ext}(X)$: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. $X$: uniform random integer variable $0, \ldots, 11$.
5. $\text{Ext}(x)$: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between $8$ and $11$?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...
   \[
   \Pr \left[ \text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2 \right] = \frac{1}{4},
   \]
Extracting randomness...

1. **$X$**: uniform random integer variable out of $0, \ldots, 7$.
2. **$\text{Ext}(X)$**: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. **$X$**: uniform random integer variable $0, \ldots, 11$.
5. **$\text{Ext}(x)$**: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between 8 and 11?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...
\[
\Pr\left[\text{Ext}(X) = 00 \mid \left|\text{Ext}(X)\right| = 2\right] = \frac{1}{4}.
\]
Extracting randomness...

1. \(X\): uniform random integer variable out of \(0, \ldots, 7\).
2. \(\text{Ext}(X)\): binary representation of \(x\).
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. \(X\): uniform random integer variable \(0, \ldots, 11\).
5. \(\text{Ext}(x)\): output the binary representation for \(x\) if \(0 \leq x \leq 7\).
6. If \(x\) is between 8 and 11?
7. Idea... Output binary representation of \(x - 8\) as a two bit number.
8. A valid extractor...
\[
\Pr\left[\text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2\right] = \frac{1}{4},
\]
Extracting randomness...

1. \( X \): uniform random integer variable out of \( 0, \ldots, 7 \).
2. \( \text{Ext}(X) \): binary representation of \( x \).
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. \( X \): uniform random integer variable \( 0, \ldots, 11 \).
5. \( \text{Ext}(x) \): output the binary representation for \( x \) if \( 0 \leq x \leq 7 \).
6. If \( x \) is between 8 and 11?
7. Idea... Output binary representation of \( x - 8 \) as a two bit number.
8. A valid extractor...
   \[
   \Pr \left[ \text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2 \right] = \frac{1}{4},
   \]
Extracting randomness...

1. $X$: uniform random integer variable out of $0, \ldots, 7$.
2. $\text{Ext}(X)$: binary representation of $x$.
3. Definition more subtle... all extracted sequence of the same length would have the same probability.
4. $X$: uniform random integer variable $0, \ldots, 11$.
5. $\text{Ext}(x)$: output the binary representation for $x$ if $0 \leq x \leq 7$.
6. If $x$ is between 8 and 11?
7. Idea... Output binary representation of $x - 8$ as a two bit number.
8. A valid extractor...

$$\Pr[\text{Ext}(X) = 00 \mid |\text{Ext}(X)| = 2] = \frac{1}{4},$$
The following is obvious, but we provide a proof anyway.

**Lemma**

*Let* \( \frac{x}{y} \) *be a faction, such that* \( \frac{x}{y} < 1 \). *Then, for any* \( i \), *we have*

\[
\frac{x}{y} < \frac{x + i}{y + i}.
\]

**Proof.**

We need to prove that \( x(y + i) - (x + i)y < 0 \). The left size is equal to \( i(x - y) \), but since \( y > x \) (as \( \frac{x}{y} < 1 \)), this quantity is negative, as required.
Theorem

Suppose that the value of a random variable $X$ is chosen uniformly at random from the integers $\{0, \ldots, m - 1\}$. Then there is an extraction function for $X$ that outputs on average at least $\lfloor \lg m \rfloor - 1 = \lfloor H(X) \rfloor - 1$ independent and unbiased bits.
**Proof**

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

Example:

2. decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks whose sizes are powers of 2.

3. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

Example: \( x = 10 \):
then falls into block \( 2^2 \)... \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.
1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

3. decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \):
then falls into block \( 2^2 \)...
\( x \) relative location is 2. Output 2 written using two bits, Output: “10”. 
**Proof**

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

3. decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \): then falls into block \( 2^2 \)... \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.
Proof

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|
| o | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

3. \( \{0, \ldots, m - 1\} \) decomposed into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \):

   then falls into block \( 2^2 \).

   \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.

Sariel (UIUC) CS573 Fall 2013 24 / 28
**Proof**

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

3. Decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \):

   then falls into block \( 2^2 \ldots \)

   \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.
Proof

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

3. decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \):
   then falls into block \( 2^2 \)...
   \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.
Proof

1. \( m \): A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. Example:

3. decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.

4. If \( x \) is in block \( 2^k \), output its relative location in the block in binary representation.

5. Example: \( x = 10 \): then falls into block \( 2^2 \)...

   \( x \) relative location is 2. Output 2 written using two bits, Output: “10”.

Sariel (UIUC) CS573 Fall 2013 24 / 28
**Proof**

1. **m**: A sum of unique powers of 2, namely \( m = \sum_i a_i 2^i \), where \( a_i \in \{0, 1\} \).

2. **Example**: 
   
   \[
   \begin{array}{cccccccccccc}
   0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 \\
   \end{array}
   \]

3. **decomposed \( \{0, \ldots, m - 1\} \) into disjoint union of blocks sizes are powers of 2.**

4. **If** \( x \) **is in block** \( 2^k \), **output its relative location in the block in binary representation.**

5. **Example**: \( x = 10 \): 
   
   \[
   \begin{array}{cccccccccccc}
   0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 \\
   \end{array}
   \]
   
   then falls into block \( 2^2 \)...

   \( x \) **relative location is 2. Output** \( 2 \) **written using two bits, Output: “10”.


Proof continued

1. Valid extractor...
2. Theorem holds if \( m \) is a power of two. Only one block.
3. \( m \) not a power of 2...
4. \( X \) falls in block of size \( 2^k \): then output \( k \) complete random bits... 
... entropy is \( k \).
5. Let \( 2^k < m < 2^{k+1} \) biggest block.
6. \( u = \lfloor \lg(m - 2^k) \rfloor < k \).
There must be a block of size \( u \) in the decomposition of \( m \).
7. two blocks in decomposition of \( m \): sizes \( 2^k \) and \( 2^u \).
8. Largest two blocks...
9. \( 2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0 \).
10. \( Y \): random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...
2. Theorem holds if $m$ is a power of two. Only one block.
3. $m$ not a power of 2...
4. $X$ falls in block of size $2^k$: then output $k$ complete random bits.
   ... entropy is $k$.
5. Let $2^k < m < 2^{k+1}$ biggest block.
6. $u = \lfloor \log_2(m - 2^k) \rfloor < k$.
   There must be a block of size $u$ in the decomposition of $m$.
7. two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.
8. Largest two blocks...
9. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.
10. $Y$: random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...

2. Theorem holds if \( m \) is a power of two. Only one block.

3. \( m \) not a power of 2...

4. \( X \) falls in block of size \( 2^k \): then output \( k \) complete random bits. ... entropy is \( k \).

5. Let \( 2^k < m < 2^{k+1} \) biggest block.

6. \( u = \left\lfloor \log_2(m - 2^k) \right\rfloor < k. \)

   There must be a block of size \( u \) in the decomposition of \( m \).

7. two blocks in decomposition of \( m \): sizes \( 2^k \) and \( 2^u \).

8. Largest two blocks...

9. \( 2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0. \)

10. \( Y \): random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...
2. Theorem holds if \( m \) is a power of two. Only one block.
3. \( m \) not a power of 2...
4. \( X \) falls in block of size \( 2^k \): then output \( k \) complete random bits. ... entropy is \( k \).
5. Let \( 2^k < m < 2^{k+1} \) biggest block.
6. \( u = \left\lfloor \log_2(m - 2^k) \right\rfloor < k \).
   There must be a block of size \( u \) in the decomposition of \( m \).
7. Two blocks in decomposition of \( m \): sizes \( 2^k \) and \( 2^u \).
8. Largest two blocks...
9. \( 2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0 \).
10. \( Y \): random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...

2. Theorem holds if $m$ is a power of two. Only one block.

3. $m$ not a power of 2...

4. $X$ falls in block of size $2^k$: then output $k$ complete random bits. ... entropy is $k$.

5. Let $2^k < m < 2^{k+1}$ biggest block.

6. $u = \left\lfloor \lg (m - 2^k) \right\rfloor < k$.

7. There must be a block of size $u$ in the decomposition of $m$.

8. Two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.

9. Largest two blocks...

10. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.

10. $Y$: random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...

2. Theorem holds if \( m \) is a power of two. Only one block.

3. \( m \) not a power of 2...

4. \( X \) falls in block of size \( 2^k \): then output \( k \) complete random bits.

... entropy is \( k \).

5. Let \( 2^k < m < 2^{k+1} \) biggest block.

6. \( u = \left\lfloor \lg(m - 2^k) \right\rfloor < k \).

There must be a block of size \( u \) in the decomposition of \( m \).

7. two blocks in decomposition of \( m \): sizes \( 2^k \) and \( 2^u \).

8. Largest two blocks...

9. \( 2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0 \).

10. \( Y \): random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...
2. Theorem holds if $m$ is a power of two. Only one block.
3. $m$ not a power of 2...
4. $X$ falls in block of size $2^k$: then output $k$ complete random bits. ... entropy is $k$.
5. Let $2^k < m < 2^{k+1}$ biggest block.
6. $u = \left\lfloor \lg(m - 2^k) \right\rfloor < k$.
   There must be a block of size $u$ in the decomposition of $m$.
7. two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.
8. Largest two blocks...
9. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.
10. $Y$: random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...
2. Theorem holds if $m$ is a power of two. Only one block.
3. $m$ not a power of 2...
4. $X$ falls in block of size $2^k$: then output $k$ complete random bits.
   ... entropy is $k$.
5. Let $2^k < m < 2^{k+1}$ biggest block.
6. $u = \left\lfloor \log(m - 2^k) \right\rfloor < k$.
   There must be a block of size $u$ in the decomposition of $m$.
7. two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.
8. Largest two blocks...
9. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.
10. $Y$: random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...
2. Theorem holds if \( m \) is a power of two. Only one block.
3. \( m \) not a power of \( 2 \)...
4. \( X \) falls in block of size \( 2^k \): then output \( k \) complete random bits.
   ... entropy is \( k \).
5. Let \( 2^k < m < 2^{k+1} \) biggest block.
6. \( u = \lfloor \lg(m - 2^k) \rfloor < k \).
   There must be a block of size \( u \) in the decomposition of \( m \).
7. two blocks in decomposition of \( m \): sizes \( 2^k \) and \( 2^u \).
8. Largest two blocks...
9. \( 2^k + 2 \cdot 2^u > m \implies 2^{u+1} + 2^k - m > 0 \).
10. \( Y \): random variable = number of bits output by extractor.
Proof continued

1. Valid extractor...

2. Theorem holds if $m$ is a power of two. Only one block.

3. $m$ not a power of 2...

4. $X$ falls in block of size $2^k$: then output $k$ complete random bits.. ... entropy is $k$.

5. Let $2^k < m < 2^{k+1}$ biggest block.

6. $u = \left\lfloor \lg(m - 2^k) \right\rfloor < k$.
   There must be a block of size $u$ in the decomposition of $m$.

7. two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.

8. Largest two blocks...

9. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.

10. $Y$: random variable = number of bits output by extractor.
1. Valid extractor...
2. Theorem holds if $m$ is a power of two. Only one block.
3. $m$ not a power of $2$...
4. $X$ falls in block of size $2^k$: then output $k$ complete random bits. ...
   ... entropy is $k$.
5. Let $2^k < m < 2^{k+1}$ biggest block.
6. $u = \lfloor \lg(m - 2^k) \rfloor < k$.
   There must be a block of size $u$ in the decomposition of $m$.
7. two blocks in decomposition of $m$: sizes $2^k$ and $2^u$.
8. Largest two blocks...
9. $2^k + 2 \times 2^u > m \implies 2^{u+1} + 2^k - m > 0$.
10. $Y$: random variable = number of bits output by extractor.
Proof continued

1. By lemma, since \( \frac{m-2^k}{m} < 1 \):

\[
\frac{m - 2^k}{m} \leq \frac{m - 2^k + (2^{u+1} + 2^k - m)}{m} + (2^{u+1} + 2^k - m) = \frac{2^{u+1}}{2^{u+1} + 2^k}.
\]

2. By induction (assumed holds for all numbers smaller than \( m \)):

\[
E[Y] \geq \frac{2^k}{m} + \frac{m - 2^k}{m} \left( \left\lfloor \log_2(m - 2^k) \right\rfloor - 1 \right) = \frac{2^k}{m} + \frac{m - 2^k}{m} (k - k + u - 1) = k + \frac{m - 2^k}{m} (u - k - 1)
\]
By lemma, since \( \frac{m - 2^k}{m} < 1 \):

\[
\frac{m - 2^k}{m} \leq \frac{m - 2^k + (2^{u+1} + 2^k - m)}{m + (2^{u+1} + 2^k - m)} = \frac{2^{u+1}}{2^{u+1} + 2^k}.
\]

By induction (assumed holds for all numbers smaller than \( m \)):

\[
E[Y] \geq \frac{2^k}{m} k + \frac{m - 2^k}{m} \left( \left\lfloor \log(m - 2^k) \right\rfloor - 1 \right)
\]

\[
= \frac{2^k}{m} k + \frac{m - 2^k}{m} (k - k + u - 1)
\]

\[
= k + \frac{m - 2^k}{m} (u - k - 1)
\]
1. By lemma, since \( \frac{m - 2^k}{m} < 1 \):

\[
\frac{m - 2^k}{m} \leq \frac{m - 2^k + (2^{u+1} + 2^k - m)}{m} + (2^{u+1} + 2^k - m) = \frac{2^{u+1}}{2^{u+1} + 2^k}.
\]

2. By induction (assumed holds for all numbers smaller than \( m \)):

\[
E[Y] \geq \frac{2^k}{m}k + \frac{m - 2^k}{m} \left( \left\lfloor \log(m - 2^k) \right\rfloor - 1 \right)
\]

\[
= \frac{2^k}{m}k + \frac{m - 2^k}{m} (k - k + u - 1)
\]

\[
= k + \frac{m - 2^k}{m} (u - k - 1)
\]
Proof continued

1. By lemma, since \( \frac{m-2^k}{m} < 1 \):

\[
\frac{m - 2^k}{m} \leq \frac{m - 2^k + (2^{u+1} + 2^k - m)}{m} + (2^{u+1} + 2^k - m) = \frac{2^{u+1}}{2^{u+1} + 2^k}.
\]

2. By induction (assumed holds for all numbers smaller than \( m \)):

\[
E[Y] \geq \frac{2^k}{m} k + \frac{m - 2^k}{m} \left( \left\lfloor \log_2(m - 2^k) \right\rfloor - 1 \right)
\]

\[
= \frac{2^k}{m} k + \frac{m - 2^k}{m} (k - k + u - 1) = 0
\]

\[
= k + \frac{m - 2^k}{m} (u - k - 1)
\]
We have:

\[ E[Y] \geq k + \frac{m - 2^k}{m} (u - k - 1) \]

\[ \geq k + \frac{2^{u+1}}{2^{u+1} + 2^k} (u - k - 1) \]

\[ = k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u) , \]

since \( u - k - 1 \leq 0 \) as \( k > u \).

2. If \( u = k - 1 \), then \( E[Y] \geq k - \frac{1}{2} \cdot 2 = k - 1 \), as required.

3. If \( u = k - 2 \) then \( E[Y] \geq k - \frac{1}{3} \cdot 3 = k - 1. \)
Proof continued..

1. We have:

$$E[Y] \geq k + \frac{m - 2^k}{m} (u - k - 1)$$

$$\geq k + \frac{2^{u+1}}{2^{u+1} + 2^k} (u - k - 1)$$

$$= k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u),$$

since $u - k - 1 \leq 0$ as $k > u$.

2. If $u = k - 1$, then $E[Y] \geq k - \frac{1}{2} \cdot 2 = k - 1$, as required.

3. If $u = k - 2$ then $E[Y] \geq k - \frac{1}{3} \cdot 3 = k - 1$. 

We have:

$$E[Y] \geq k + \frac{m - 2^k}{m} (u - k - 1)$$

$$\geq k + \frac{2^{u+1}}{2^{u+1} + 2^k} (u - k - 1)$$

$$= k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u),$$

since $u - k - 1 \leq 0$ as $k > u$.

If $u = k - 1$, then $E[Y] \geq k - \frac{1}{2} \cdot 2 = k - 1$, as required.

If $u = k - 2$ then $E[Y] \geq k - \frac{1}{3} \cdot 3 = k - 1$. 
Proof continued..

1. We have:

\[ E[Y] \geq k + \frac{m - 2^k}{m} (u - k - 1) \]

\[ \geq k + \frac{2^{u+1}}{2^{u+1} + 2^k} (u - k - 1) \]

\[ = k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u) , \]

since \( u - k - 1 \leq 0 \) as \( k > u \).

2. If \( u = k - 1 \), then \( E[Y] \geq k - \frac{1}{2} \cdot 2 = k - 1 \), as required.

3. If \( u = k - 2 \) then \( E[Y] \geq k - \frac{1}{3} \cdot 3 = k - 1 \).
1. \( \mathbb{E}[Y] \geq k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u) \).
   And \( u - k - 1 \leq 0 \) as \( k > u \).

2. If \( u < k - 2 \) then

   \[
   \mathbb{E}[Y] \geq k - \frac{2^{u+1}}{2^k} (1 + k - u)
   = k - \frac{k - u + 1}{2^{k-u-1}}
   = k - \frac{2 + (k - u - 1)}{2^{k-u-1}}
   \geq k - 1,
   \]

   since \( (2 + i)/2^i \leq 1 \) for \( i \geq 2 \).
1. \[ E[Y] \geq k - \frac{2^{u+1}}{2^{u+1} + 2^k} (1 + k - u). \]
   And \( u - k - 1 \leq 0 \) as \( k > u \).

2. If \( u < k - 2 \) then

   \[ E[Y] \geq k - \frac{2^{u+1}}{2^k} (1 + k - u) \]
   
   \[ = k - \frac{k - u + 1}{2^{k-u-1}} \]
   
   \[ = k - \frac{2 + (k - u - 1)}{2^{k-u-1}} \]
   
   \[ \geq k - 1, \]

   since \( (2 + i) / 2^i \leq 1 \) for \( i \geq 2 \).