



アルゴリズムとデータ構造III

14回目: 1月29日(木)

テキスト圧縮 (zip),
音声圧縮 (ADPCM, MP3, CELP),
画像圧縮 (JPEG)

授業資料 <http://ir.cs.yamanashi.ac.jp/~ysuzuki/algorithm3/index.html>

授業の予定(中間試験まで)

1	10/02	スタック(後置記法で書かれた式の計算)
2	10/09	チューリング機械, 文脈自由文法
3	10/16	構文解析 CYK法
4	10/23	構文解析 CYK法
5	10/30	構文解析(チャート法), グラフ(ダイクストラ法)
6	11/06	構文解析(チャート法), グラフ(ダイクストラ法, DPマッチング)
7	11/13	グラフ(DPマッチング, A*アルゴリズム)
8	11/20	グラフ(A*アルゴリズム), 前半のまとめ
9	11/27	中間試験

授業の予定(中間試験以降)

10	12/04	全文検索アルゴリズム (simple search, KMP)
11	12/18	全文検索アルゴリズム (BM, Aho-Corasick)
12	01/08	全文検索アルゴリズム (Aho-Corasick), データ圧縮
13	01/15	暗号 (黄金虫, 踊る人形) 符号化 (モールス信号, Zipfの法則, ハフマン符号) テキスト圧縮
14	01/29 B2-11	テキスト圧縮 (zip), 音声圧縮 (ADPCM, MP3, CELP), 画像圧縮 (JPEG)
15	02/05	期末試験 T1-31



期末試験

- 日時：2月5日（木）2時限
- 教室：T1-31



特別試験(予定)

- 3月3日(火) 学習日
- 3月4日(水) 試験日

- 対象者にはCNSで連絡



本日のメニュー

- Zipfの法則
- 暗号
 - 黄金虫 (The gold bug)
 - 踊る人形 (The Adventure of the Dancing Men)
- 符号化
 - モールス信号
 - ハフマン符号
- テキスト圧縮
- テキスト圧縮 (zip)
- 音声圧縮 ADPCM, MP3, 音声圧縮 (CELP)
- 画像圧縮 (JPEG)



データ圧縮

- 対象データ

- テキスト
- 音声
 - 音楽
 - 話し声
- 画像
- 動画

- 圧縮方式

- 可逆圧縮(ロスレス圧縮)
- 非可逆圧縮(ロッキー圧縮)

ジップの法則(Zipf's law)

経験則:「あるタイプの現象が生起する確率はその現象の生起する順位に反比例する」

- Zipfの法則が当てはまる事象
 - 文字毎の出現頻度
 - コンピュータにおけるコマンドの使用頻度
 - Webページのアクセス頻度
 - 都市の人口
 - 文献の参照回数
 - 会社でのランク(役職)と給料など
 - ケータイのシェア(docomo, au, softbank, e-mobile)



会社でのランク(役職)と給料

ランク	役職	給料	割合(確率)
1	社長	2,000,000	0.54
2	副社長	1,000,000	0.27
3	部長	700,000	0.19
.....

- 「あるタイプの現象が生起する確率はその現象の生起する順位に反比例する」

携帯電話：各グループ毎の加入者数累計 (2008年12月 ケータイWatchより)

順位	事業者	累計	割合(確率)	Zipf's law C=0.52
1	NTTドコモ	54,155,100	51.2%	52.0%
2	KDDI	30,550,200	28.9%	26.0%
3	ソフトバンク	19,999,800	18.9%	17.3%
4	イー・モバイル	1,120,100	1.1%	13.0%

- 「あるタイプの現象が生起する確率はその現象の生起する順位に反比例する」

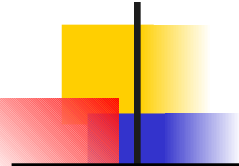


自然言語の統計的性質

- 文字の使用頻度(英語) _はスペース

順位	文字	%	2	%	3	%	4	%
1	_	17.4	e_	3.0	_th	1.6	_the	1.2
2	e	9.7	_t	2.4	the	1.3	the_	1.0
3	t	7.0	th	2.0	he_	1.3	_of_	0.6
4	a	6.1	he	1.9	_of	0.6	and_	0.4
5	o	5.9	_a	1.7	of_	0.6	_and	0.4
6	i	5.5	s_	1.7	ed_	0.5	_to_	0.4
7	n	5.5	d_	1.5	_an	0.5	ing_	0.3

単語の使用頻度



順位	単語	%	2	%	3	%
1	the	6.1	of the	0.9	one of the	0.03
2	of	3.5	in the	0.5	as well as	0.02
3	and	2.7	to the	0.3	the United States	0.02
4	to	2.5	on the	0.2	out of the	0.02
5	a	2.1	and the	0.2	some of the	0.01
6	in	1.9	for the	0.1	the end of	0.01
7	that	0.9	to be	0.1	the fact that	0.01



単語の出現頻度分布

- ジップの法則 (Zipf's law) を単語の出現頻度に適用すると



単語の出現頻度分布

- ジップの法則(Zipf's law): 単語の出現順位 (r) と出現頻度 (f) は反比例の関係にある

$$r = \frac{C}{f} \quad f = \frac{C}{r}$$

n 番目の単語の出現確率 P_n

$$P_n = \frac{C}{n}$$

C は定数

低頻度の語には当てはまらない

データの頻度分布の偏りを利用 した技術



- 暗号(換字式)の復号
- データ圧縮(ロスレス)
 - キー入力時の打鍵回数の削減
- 検索アルゴリズム(Boyer-Moore)



小説中での暗号解読の解説

- 黄金虫 (The gold bug)
 - 著者: エドガー・アラン・ポー
 - 作品: 翻訳版
 - <http://www.aozora.gr.jp/cards/000094/card2525.html>
 - 作品: 原文
 - http://en.wikisource.org/wiki/The_Gold-Bug
- 踊る人形 (The Adventure of the Dancing Men)
 - 著者: アーサー・コナン・ドイル
 - 作品: 翻訳版 題: 暗号舞踏人の謎
 - <http://www.aozora.gr.jp/cards/000009/card45340.html>
 - 作品: 原文
 - http://en.wikisource.org/wiki/The_Adventure_of_the_Dancing_Men

黄金虫(エドガー・アラン・ポー) に出てくる暗号(換字式)

- 暗号解読の解説
- 暗号は多分英語
- 英語は文字によって出現確率が違う
 - 出現確率の高い方から並べると
 - eaoidhnrstuycfglmwbkpxz
 - eは頻出, eeも頻出
 - theも頻出
- 対応がとれた文字は置き換え, 前後の文字を推理する

携帯電話の文字キー

あ か さ

た な は

ま や ら

わ

abc def

ghi jkl mno

pqrs tuv wxyz

-

■ かなキー

- 覚えやすい, 直感的な並べ方
- 文字毎の出現確率を調べることで, キーを押す回数を減らすことが出来る.

■ アルファベットキー

- アルファベット順に26文字を8つのキーに割り振っている
- pqrsとwxyzは4文字を1つのキーに割り振られている

おまけ

Scrabble (英単語作成ボードゲーム)の得点

■ Scrabble

- 対戦型英単語作成ゲーム
- ボード上に手持ちの文字をならべ英単語を作成
- 作成した単語の文字に書かれている得点を合計し、高得点を競う
- 英単語を作りにくい文字には高得点が割り振られている。
 - 1点:E, A, I, O, R, N, T, L, S, U
 -
 - 10点:Q, Z



データ圧縮への利用

- モールス信号
- ハフマン符号

モールス信号の符号

- ・(短点)とー(長点)を用いてアルファベットを表現する
- 情報を早く送るための工夫
 - よく使われる文字(例えばe,t)は短い
 - e: ・ (短点1文字)
 - t: ー (長点1文字)
 - あまり使われない文字(例えばqは4文字)は長い
 - q: ーー・ー

モールス信号の符号

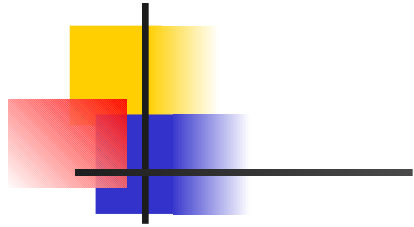
- ・(短点)と- (長点:短点3つ分の長さ)を用いてアルファベットを表現する
- 区切り記号
 - 文字の切れ目:短点3つ分の間隔
 - 単語の切れ目:短点7つ分の間隔
- L: ·-·· (LifeカードのCMに使われていた)
- SOS: ··· --- ···



ハフマン符号

- 2分木を使って文字の出現頻度順に並べる
- 葉 = 文字
- 浅い: 符号長が短い, 深い: 符号長が長い
- 平均符号長が最小になることが保証されている

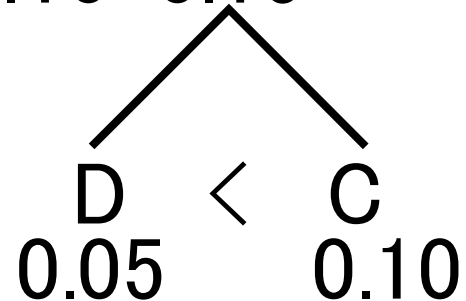
ハフマン符号の作り方 1/5



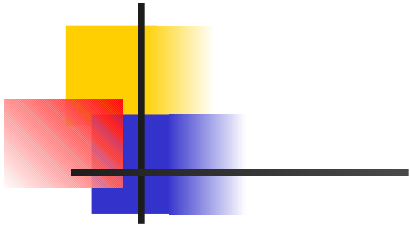
- 頻度の低い文字を2文字 (DC) を選び、頻度の低い方を左の葉、頻度の高い方を右の葉に置き、2分木をつくる。
- ルートノードには2つの葉の頻度の和を書き込む

文字	頻度
A	0.25
B	0.20
C	0.10
D	0.05
E	0.40

$$0.05 + 0.10 = 0.15$$

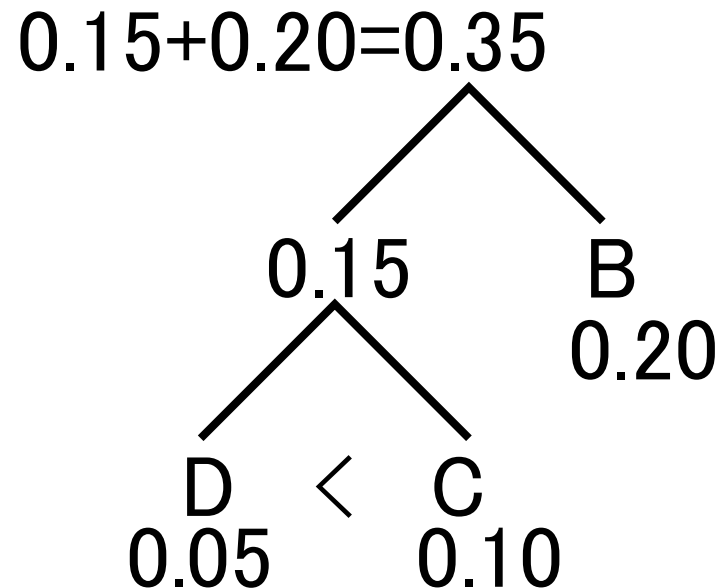


ハフマン符号の作り方 2/5

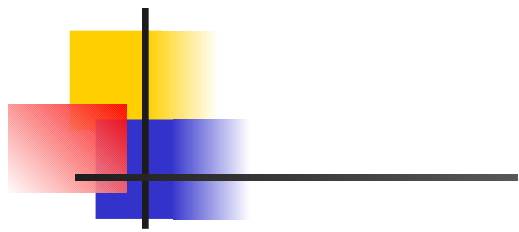


文字	頻度
A	0.25
B	0.20
(DC)	0.15
E	0.40

- (DC)統合後，頻度の低いBと(DC)連合を選ぶ. Bと(DC)連合の頻度を比較し，頻度の高いBを右ノードに，低い(DC)連合を左ノードに配置する.
- ルートノードには頻度の和を書き込む



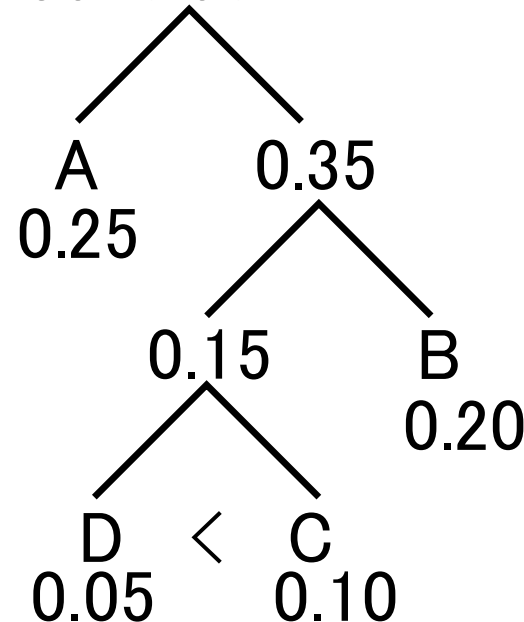
ハフマン符号の作り方 3/5



- ((DC)B)統合後, 頻度の低いAと((DC)B) 連合を選ぶ. Aと((DC)B) 連合の頻度を比較し, 頻度の高い((DC)B) 連合を右ノードに, 低いAを左ノードに配置する.
- ルートノードには頻度の和を書き込む

文字	頻度
A	0.25
((DC)B)	0.35
E	0.40

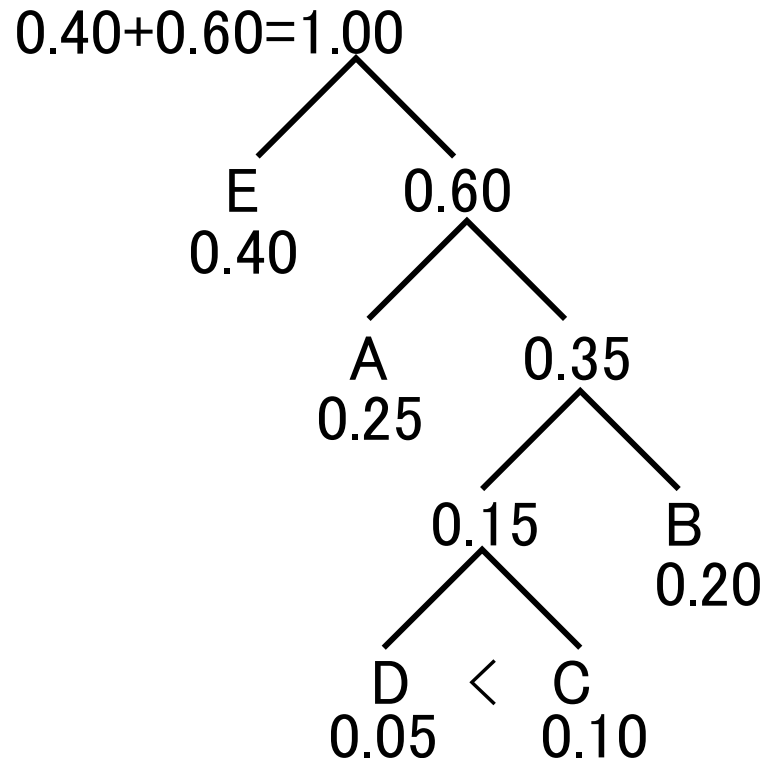
$$0.25 + 0.35 = 0.60$$



ハフマン符号の作り方 4/5

文字	頻度
(A((DC)B))	0.60
E	0.40

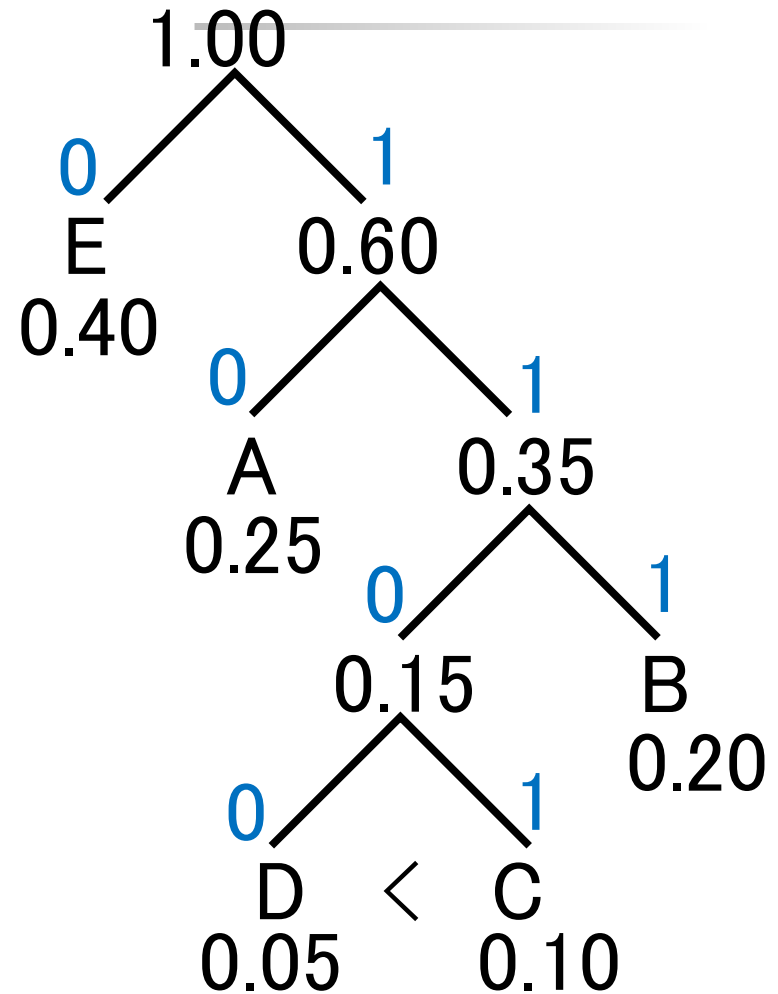
- (A((CD)B))統合後, 頻度の低いEと(A((CD)B))連合を選ぶ. Eと(A((CD)B))連合の頻度を比較し, 頻度の高い(A((CD)B))連合を右ノードに, 低いEを左ノードに配置する.
- ルートノードには頻度の和を書き込む



ハフマン符号の作り方 5/5

- 左のノードに0,
右のノードに1を付与する

文字	頻度	符号
A	0.25	10
B	0.20	111
C	0.10	1101
D	0.05	1100
E	0.40	0



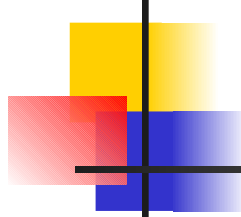


文字⇔ハフマン符号の変換

文字	頻度	符号
A	0.25	10
B	0.20	111
C	0.10	1101
D	0.05	1100
E	0.40	0

- 10111110111000
 - 10|111|1101|1100|0
 - A|B|C|D|E
- BBCEDA
 - 11111111010110010
 - 111|111|1101|0|1100|10

ASCII文字コード(8bit)からハフマン符号へ



文字	頻度	符号
A	0.25	10
B	0.20	111
C	0.10	1101
D	0.05	1100
E	0.40	0

- A
 - ASCII: 01000001 (0x41) 8bit
 - Huffman: 10 : 2bit
- E
 - ASCII: 01000101 (0x45) 8bit
 - Huffman: 0 : 1bit

練習問題1

- 下の表のような記号の出現頻度のとき, ハフマン符号をつくりなさい. 但しハフマン符号作成のための二分木も書くこと.

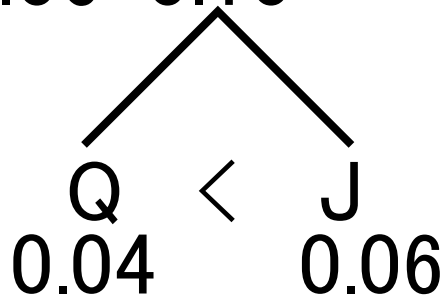
記号	頻度
B	0.17
D	0.20
E	0.33
F	0.12
J	0.06
K	0.08
Q	0.04
合計	1.00

練習問題1 解答例 1/7

- 下の表のような記号の出現頻度のとき, ハフマン符号をつくりなさい. 但しハフマン符号作成のための二分木も書くこと.

記号	頻度
B	0.17
D	0.20
E	0.33
F	0.12
J	0.06
K	0.08
Q	0.04
合計	1.00

$$0.04 + 0.06 = 0.10$$



練習問題1 解答例 1/7

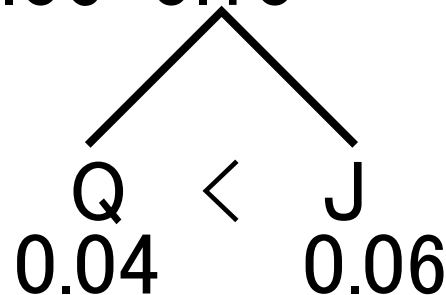
- 下の表のような記号の出現頻度のとき、ハフマン符号をつくりなさい。但しハフマン符号作成のための二分木も書くこと。

記号	頻度
B	0.17
D	0.20
E	0.33
F	0.12
J	0.06
K	0.08
Q	0.04
合計	1.00



記号	頻度
E	0.33
D	0.20
B	0.17
F	0.12
K	0.08
J	0.06
Q	0.04
合計	1.00

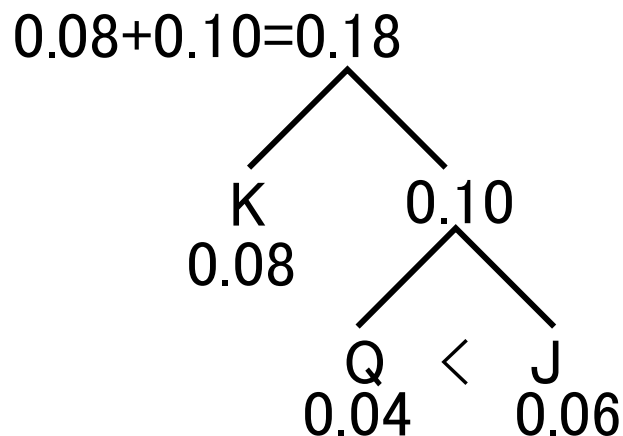
$$0.04 + 0.06 = 0.10$$



練習問題1 解答例 2/7

- 下の表のような記号の出現頻度のとき, ハフマン符号をつくりなさい. 但しハフマン符号作成のための二分木も書くこと.

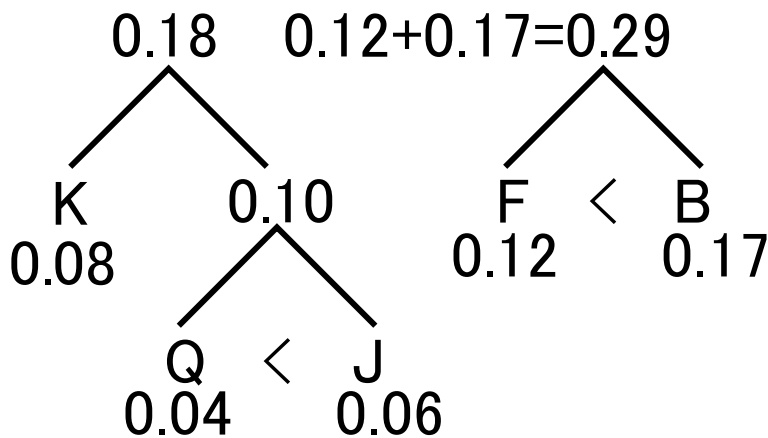
記号	頻度
E	0.33
D	0.20
B	0.17
F	0.12
(Q J)	0.10
K	0.08
合計	1.00



練習問題1 解答例 3/7

- 下の表のような記号の出現頻度のとき, ハフマン符号をつくりなさい. 但しハフマン符号作成のための二分木も書くこと.

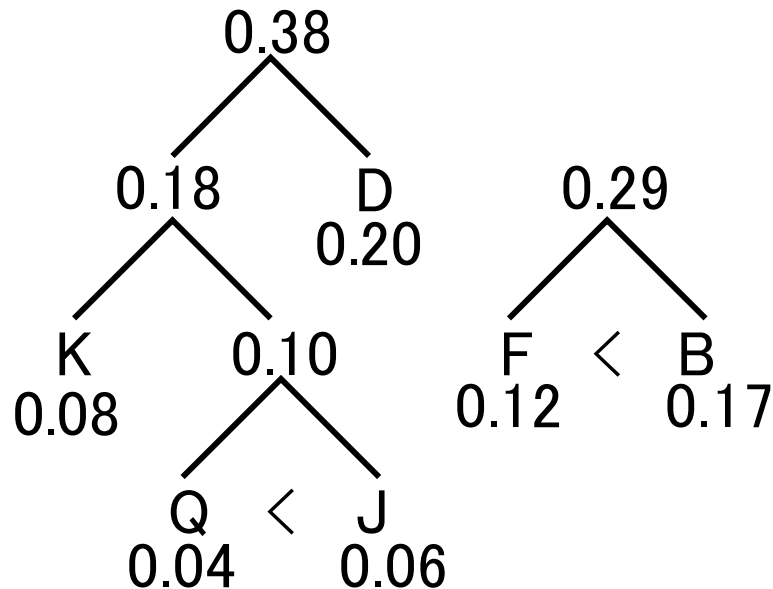
記号	頻度
E	0.33
D	0.20
(K(Q J))	0.18
B	0.17
F	0.12
合計	1.00



練習問題1 解答例 4/7

- 下の表のような記号の出現頻度のとき、ハフマン符号をつくりなさい。但しハフマン符号作成のための二分木も書くこと。

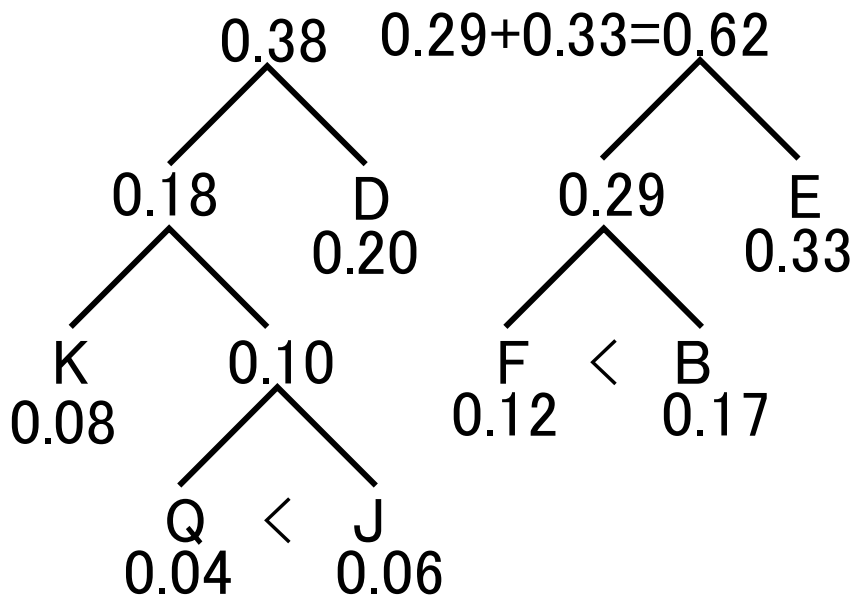
記号	頻度
E	0.33
(F B)	0.29
D	0.20
(K(Q J))	0.18
合計	1.00



練習問題1 解答例 5/7

- 下の表のような記号の出現頻度するとき、ハフマン符号をつくりなさい。但しハフマン符号作成のための二分木も書くこと。

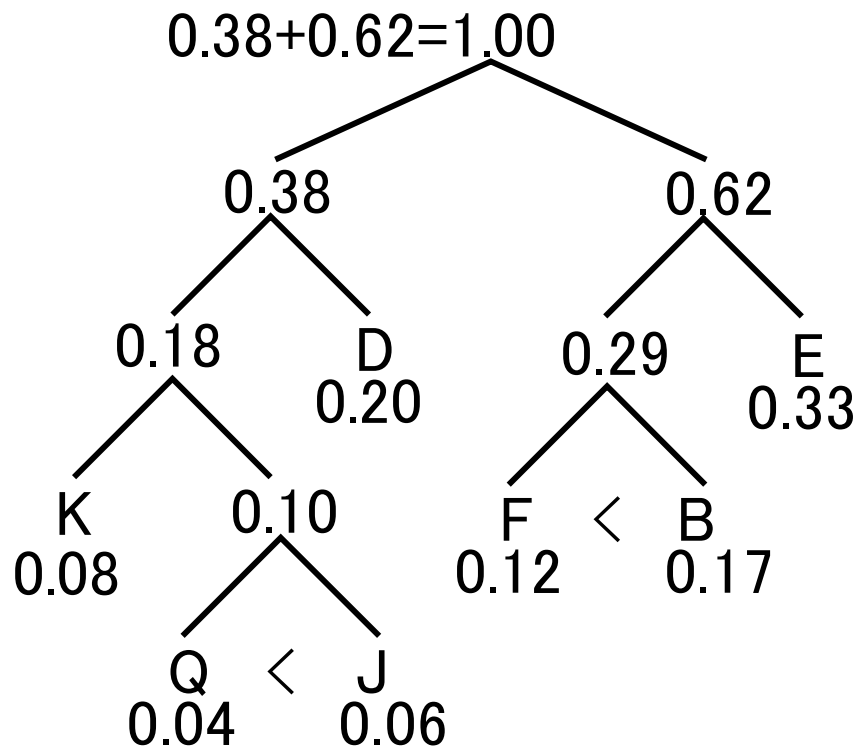
記号	頻度
((K(Q J))D)	0.38
E	0.33
(F B)	0.29
合計	1.00



練習問題1 解答例 6/7

- 下の表のような記号の出現頻度のとき, ハフマン符号をつくりなさい. 但しハフマン符号作成のための二分木も書くこと.

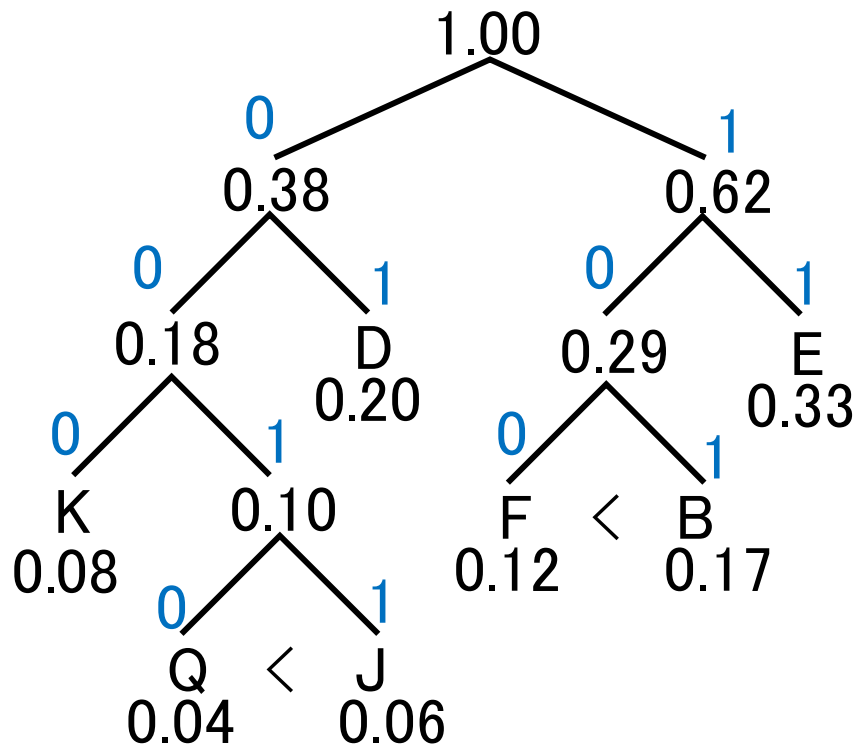
記号	頻度
((F B) E)	0.62
((K(Q J))D)	0.38
合計	1.00



練習問題1 解答例 7/7

- 下の表のような記号の出現頻度のとき、ハフマン符号をつくりなさい。但しハフマン符号作成のための二分木も書くこと。

記号	頻度	コード
B	0.17	101
D	0.20	01
E	0.33	11
F	0.12	100
J	0.06	0011
K	0.08	000
Q	0.04	0010
合計	1.00	





ハフマン符号の特徴

- 各記号がリーフノード(葉)に対応している
 - ハフマン符号列を左からトレースすることで、記号の区切りが分かる
 - 区切り記号を入れる必要がない

エントロピー(平均情報量)

(これ以上圧縮出来ない限界点)

$$H = - \sum_i P_i \log_2 P_i : P_i \text{はデータ}i\text{の出現確率}$$

エントロピー $H \leq$ ハフマンコードの平均符号長

練習問題1の例

$$\begin{aligned} H &= - \left(P_B \log_2 P_B + P_D \log_2 P_D + P_E \log_2 P_E + P_F \log_2 P_F \right. \\ &\quad \left. + P_J \log_2 P_J + P_K \log_2 P_K + P_Q \log_2 P_Q \right) \\ &= - (0.17 \log_2 0.17 + 0.20 \log_2 0.20 + 0.33 \log_2 0.33 + 0.12 \log_2 0.12 \\ &\quad + 0.06 \log_2 0.06 + 0.08 \log_2 0.08 + 0.04 \log_2 0.04) \\ &= 2.5 \end{aligned}$$

$$\text{ハフマンコードの平均符号長} = (3 + 2 + 2 + 3 + 4 + 3 + 4) / 7 = 3.0$$



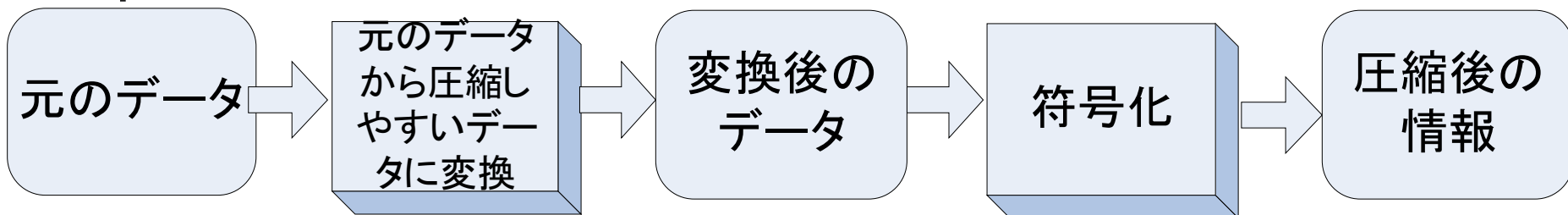
英語のエントロピー

- アルファベット26 + スペース = 27 文字が全て等確率で生じたと仮定すると、エントロピーは1文字当たり

$$-\sum_{i=1}^{27} \frac{1}{27} \log_2 \frac{1}{27} = \log_2 27 = 4.76$$

- 前出の表のような確率分布をなしている場合には、4.03 ビット/文字となる。

データ圧縮処理の流れ



圧縮しやすいデータとは：
出現する事象の確率の偏りが大きい
出現する事象の数が少ない

ほとんどの場合ハフマン符号が使用される

ハフマン符号の平均符号長がエントロピー（圧縮限界）をよく近似しているから



Zip 圧縮

- ファイル圧縮
- 可逆圧縮
- ハフマン符号化を使用

- ソフトウェア
 - PKZIP, ZIP, gzipなど

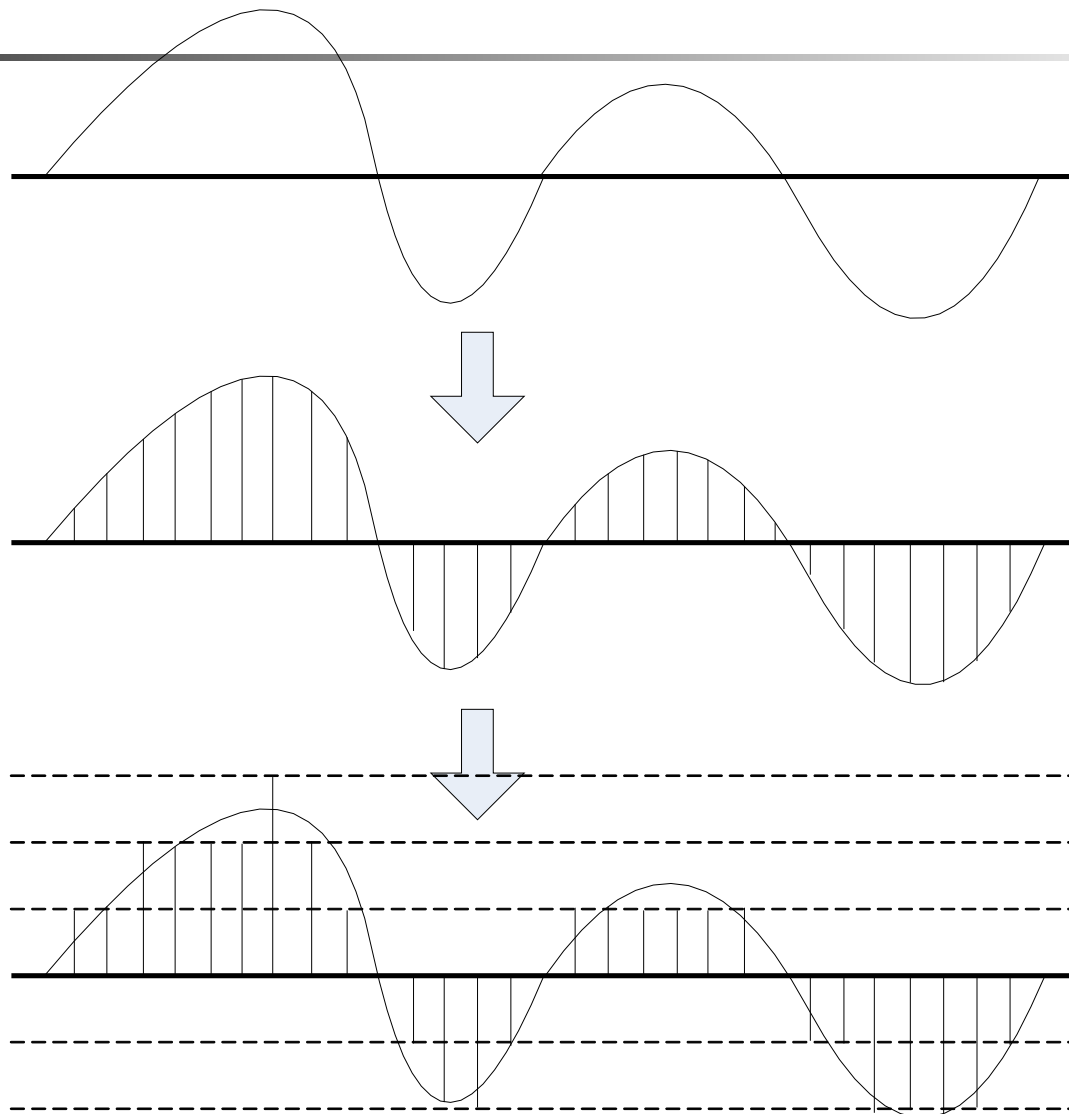


文書圧縮

- 可逆圧縮
 - ハフマン符号を使って圧縮
- 非可逆圧縮
 - 自動要約？

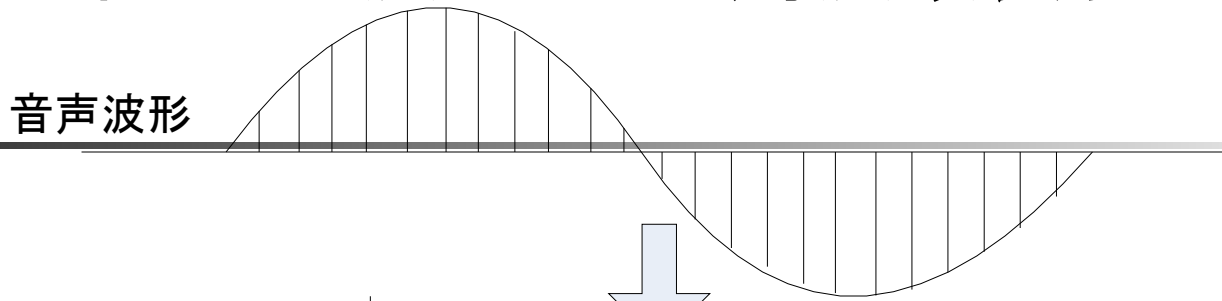
音声圧縮

- アナログ波形 → デジタル波形

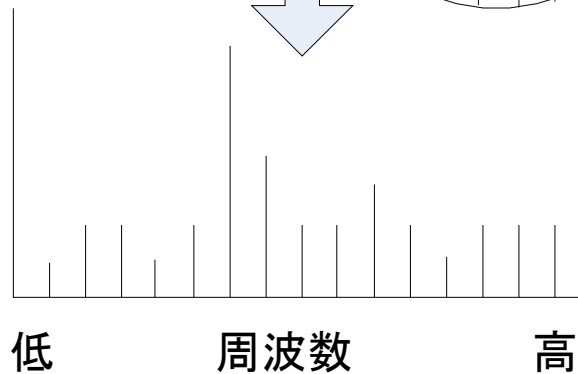


音声圧縮

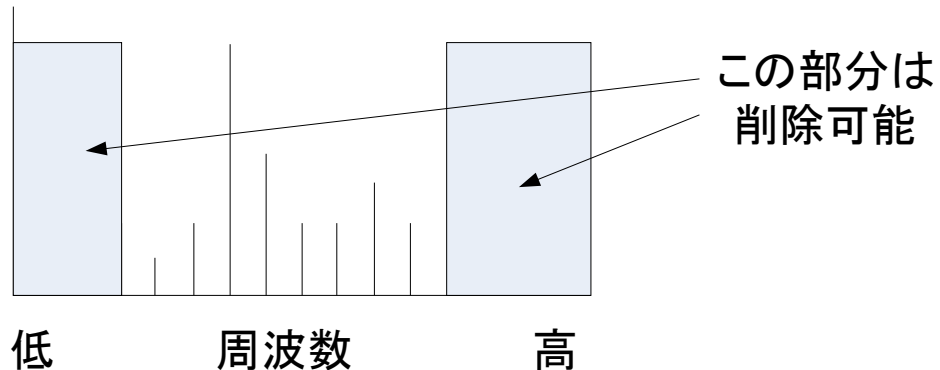
■ デジタル波形 → 周波数表現



周波数表現



周波数表現



人間が聞こえる音を再現: **MP3 128kbit/s-192kbit/s**

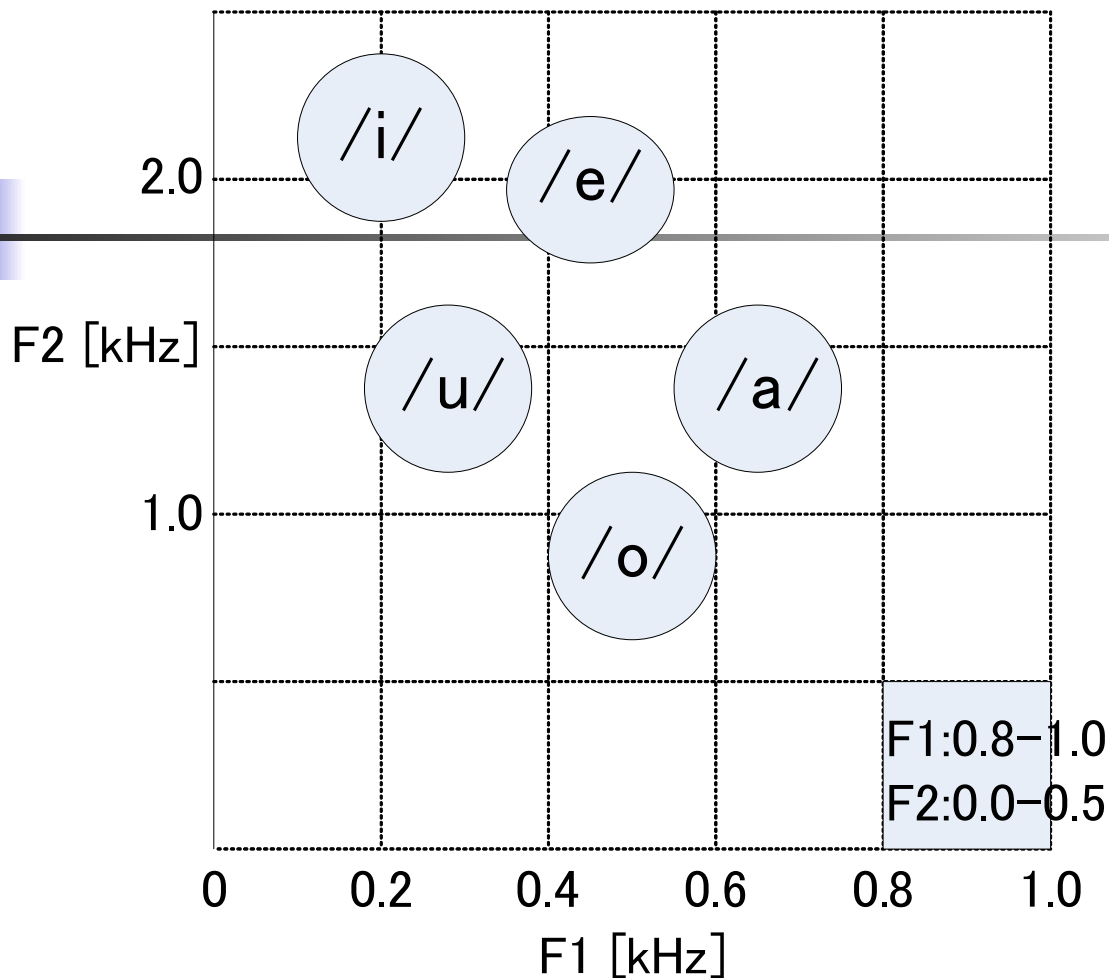
音声聞き取れればよい: **携帯電話 8kbit/s**



音声圧縮 ベクトル量子化

- 音声(人間の声)とは
 - 声帯の振動による音 → 声道の形により様々な声を発声できる
 - 声帯と声道によって出せる音は決まってしまう
- 音声の偏り
 - 話す言語(日本語, 英語)によっても限定される
 - いくつかのパラメータにより, ほぼ全ての声を決められる
 - パラメータは独立ではない(声道などの制約)
- データの頻度分布の偏りを利用して圧縮

日本語母音の第1, 第2フォルマント



日本語母音のフォルマント(男声)

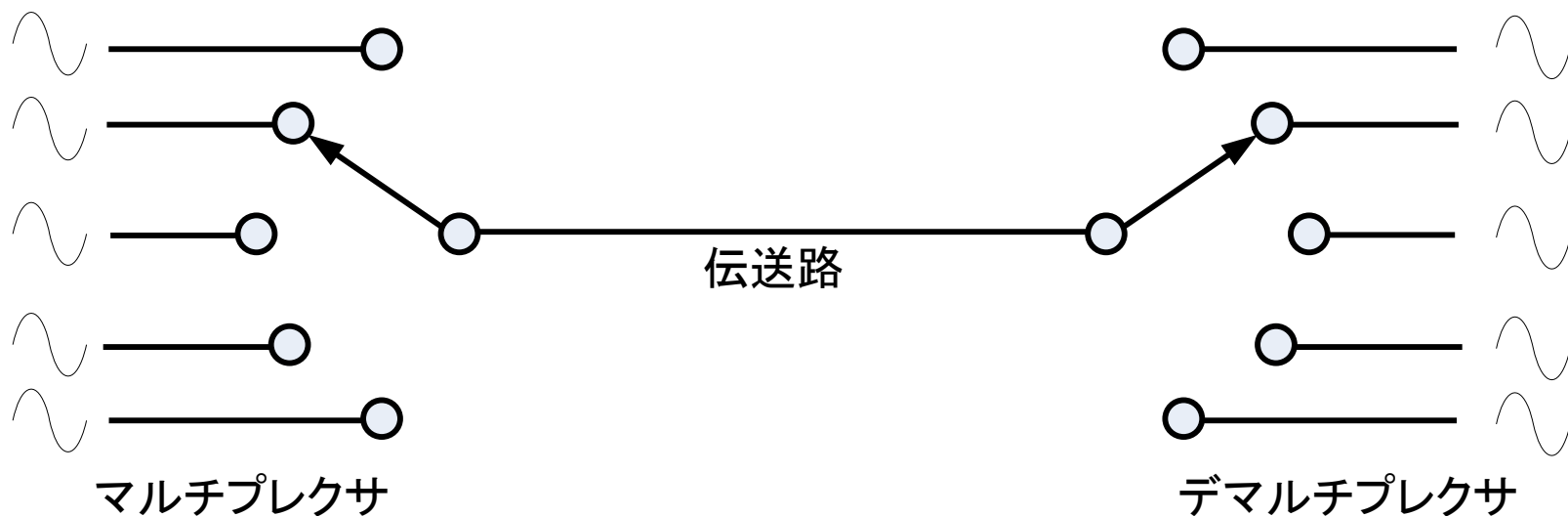
F1:0.8-1.0 F2:0.0-0.5 のような分類(クラスタ数:5x5=25)をするより
音声の特徴を活かした*/i/*, */e/*, */u/*, */o/*, */a/* のような分類のほうが圧縮しやすい

音声圧縮 PCM符号化 1/2

- PCM (Pulse Code Modulation)
- アナログ信号をデジタルデータに変換
- アナログ信号を一定時間毎に標本化し、定められたビット数の整数値に量子化する
- 音声品質を決定するもの
 - 標本化周波数
 - 量子化ビット数
- CDの音質
 - 標本化周波数: 44.1kHz
 - 量子化ビット数: 16bit

音声圧縮 PCM符号化 2/2

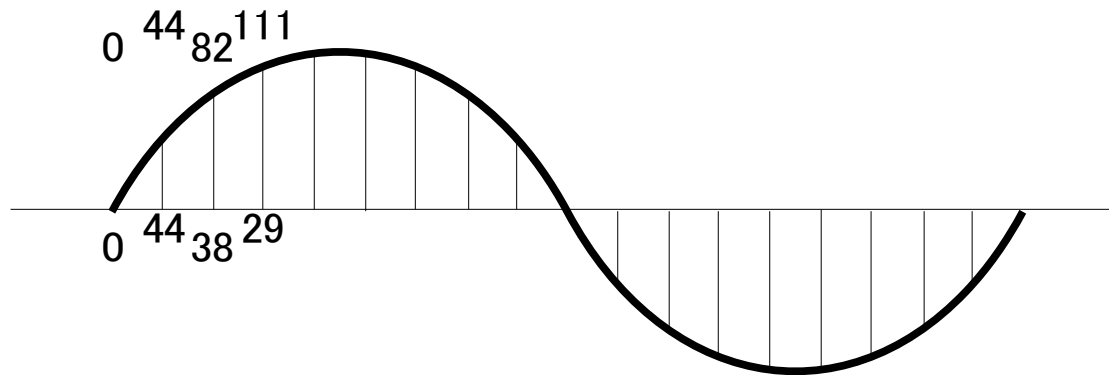
- アナログ信号のままでは1回線で複数の信号を同時に送ることは難しいが、PCMに変換すると時分割により、同時に複数の信号を送ることが出来る → 音声圧縮
- 送信側にマルチプレクサ(データセレクタ), 受信側にデマルチプレクサ(データディストリビュータ)を使う



音声圧縮 ADPCM符号化

- Adaptive Differential Pulse Code Modulation
- 音声波形は連続的に変化している。
- → 前回のサンプリングからの差分を記録するだけなら量子化ビット数を抑えられる
 - (例えば16ビットを12ビットに圧縮できる)
- → 音声圧縮できる

PCM: 128 ~ -127 (8bit)



ADPCM: 64 ~ -63 (7bit)

サンプルNo.	PCM	ADPCM
0	0	0
1	44	44
2	82	38
3	111	29
4	126	15
5	126	0
6	111	-15
7	82	-29
8	44	-38
9	0	-44
10	-44	-44
11	-82	-38
12	-111	-29
13	-126	-15
14	-126	0
15	-111	15
16	-82	29
17	-44	38
18	0	44
最大値	126	44
使用ビット数	8	7



音声圧縮 MP3

- MPEG-1オーディオ・レイヤⅢ
- シリコンオーディオプレーヤーなどで使用

音声圧縮 (CELP)

Code Excited Linear Prediction

- 携帯電話の通話の圧縮に使われている
- CS-ACELPの場合8kbps (8 kbit/s)
- ベクトル量子化と線形予測を利用

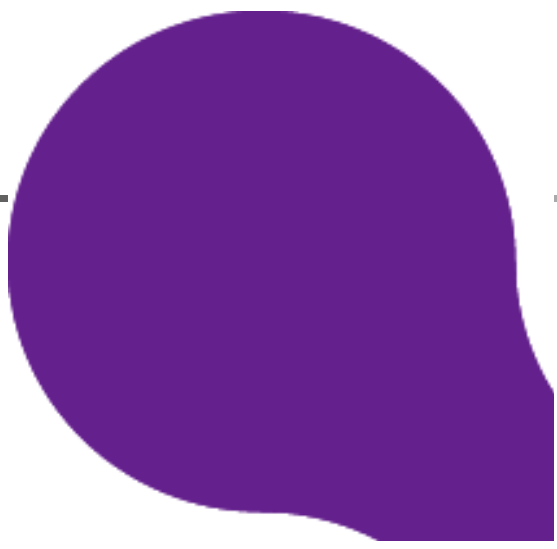


画像圧縮 (JPEG) 手法

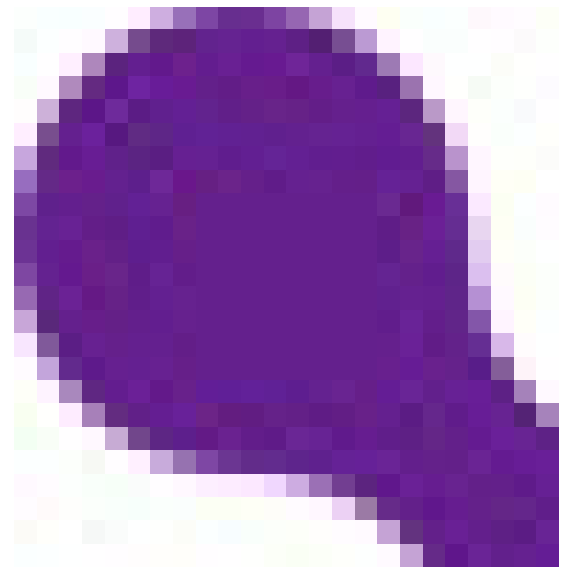
1. 画像を8x8のブロック単位に分割
2. ブロックごとにDCTを行い空間領域から周波数領域へ変換(データを低周波部分に偏在させる)
3. 高周波部分をカットし, 圧縮
4. ハフマン符号を使って符号化

JPEGとPNG

元の画像 (184KB)

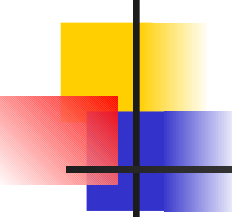


JPEG (14KB)



PNG (13KB)

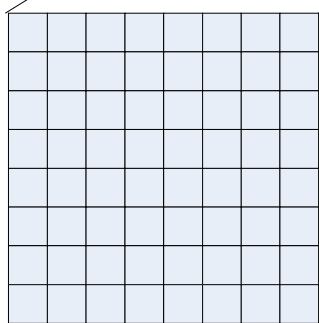




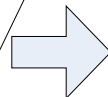
DCT (離散コサイン変換)

- JPEGなどで利用されている
- 空間領域から周波数領域へ変換
- どんな複雑な波でも三角関数の和で近似出来る (フーリエ級数展開)
- 人間の視覚 低周波に敏感 高周波には鈍感 → 低周波: 細かく量子化, 高周波: 粗く量子化
- 高周波の項の係数は0になる → データの間引きが可能

JPEGのDCT(離散コサイン変換)

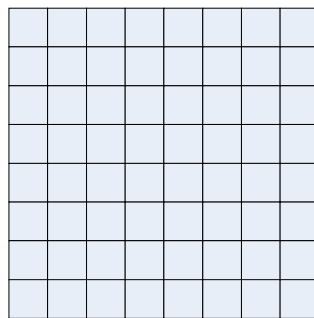


空間領域

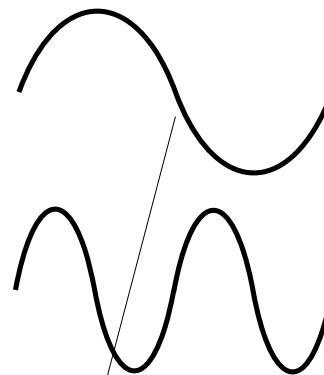


低周波
↓
高周波

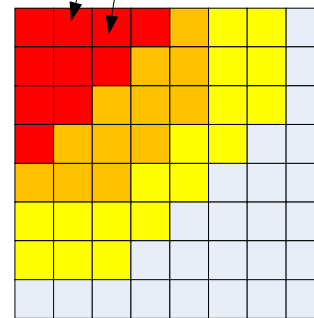
低周波 → 高周波



周波数領域



低 → 高



周波数領域

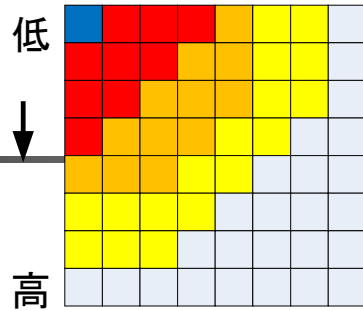
■ 係数大 → 細かく量子化

□ 係数小 → 粗く量子化

JPEGのハフマン符号化

周波数領域

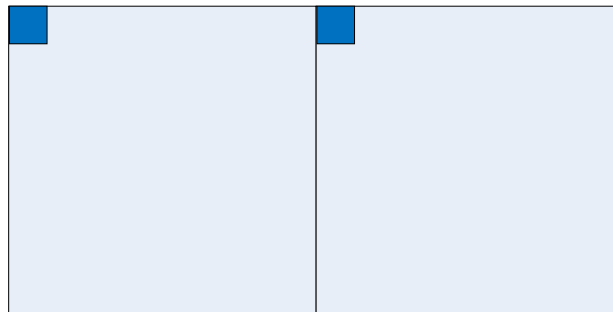
低 → 高



■ ブロック内の平均 → 直流成分

その他 → 交流成分

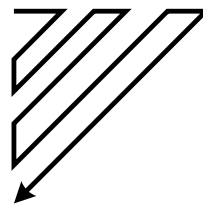
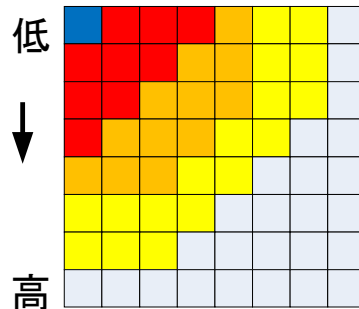
直流成分



直流成分 隣り合うブロックの直流成分との差分を量子化し、ハフマンコード化

交流成分

低 → 高



交流成分 左上から右下へジグザグにたどる順に量子化された非ゼロの係数を符号化する

周波数領域

授業のまとめ

- チューリングマシン
- スタック
- 動的計画法を利用したアルゴリズム
 - 構文解析: CKY (チャート法)
 - 最短経路探索: ダイクストラ法, A*アルゴリズム
 - マッチング: DPマッチング
- 文字列検索
 - Simple Search, KMP法, BM法, Aho-Corasick法
- データ圧縮
 - エントロピー
 - 文書, 音声, 画像の圧縮
 - ハフマン符号化
 - DCT(離散コサイン変換)