本日の実習手順

§6.5.1，§6.5.2

認識用設定ファイルの作成
単語音声学習データの作成
（ 録音→ラベル作成→切り出し ）

§6.5.3

スペクトル分析

§6.5.4

単語 HMM の学習

§6.5.5

学習の検証

§6.5.6

On-The-Fly 単語認識

メディア情報学実験メディア
情報検索・認識実験（音声）

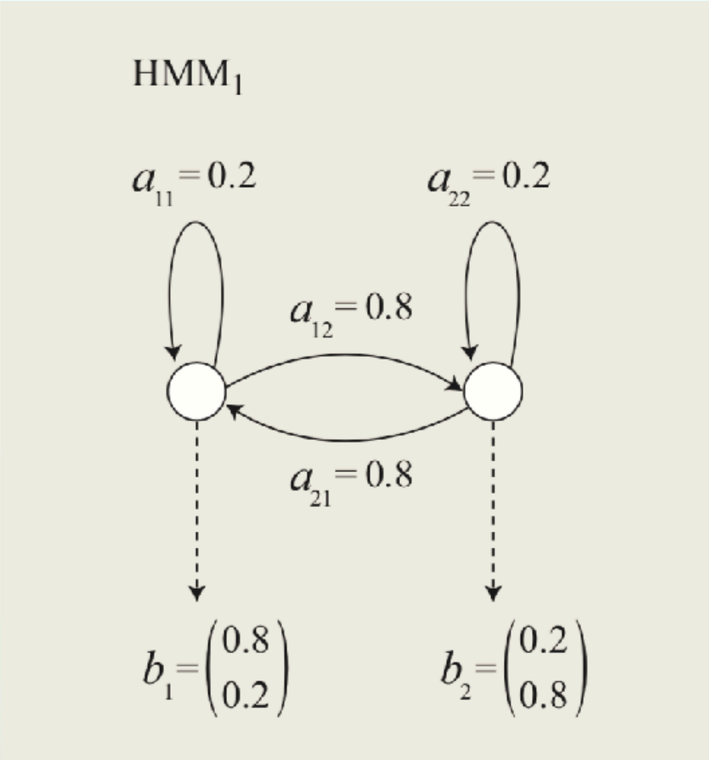
音声認識

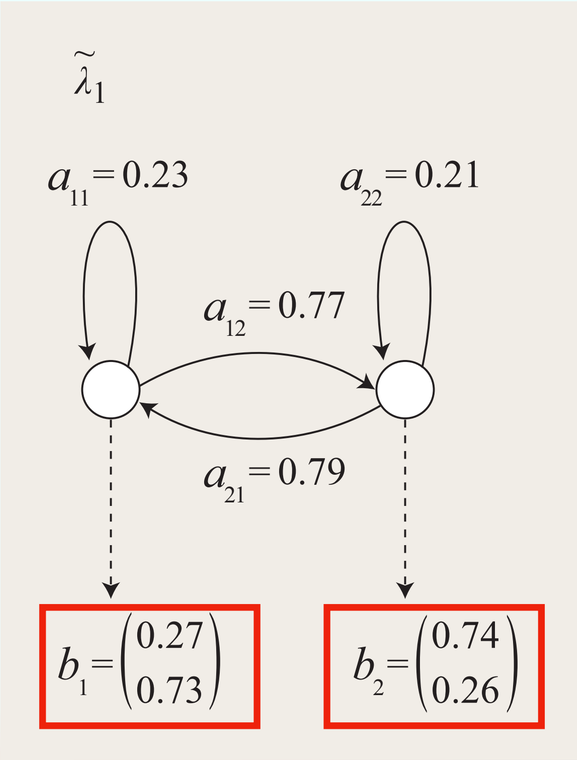
Spoken Word Recognition

出席カードに学籍番号と氏名を記入し、
2限の時間中に前方の箱に提出してください。

DAY2：HMMの学習結果状態が逆転する件

信号源





状態遷移構造が同じならば，状態出力分布が入れ替わっても同じ信号源なのでOK。

本日の実習手順

§6.5.1，§6.5.2

認識用設定ファイルの作成
単語音声学習データの作成
（ 録音→ラベル作成→切り出し ）

↓

§6.5.3

スペクトル分析

↓

§6.5.4

単語 HMM の学習

↓

§6.5.5

学習の検証

↓

§6.5.6

On-The-Fly 単語認識

単語HMM学習の要点

- **単語の前後に100msの無音声区間があるようにラベルを付ける**
- 音声波形、MFCCファイルの保存ディレクトリに注意
- 1単語当たりの学習データ数は10個丁度でなくてもよい
- タスクに応じた設定ファイル：**各自エディタで編集**

~/asr/wrecog/lib/HMMList

表示文字列、単語名、HMMの対応表

```
0 rei   hmm/rei.hmm
1 ichi  hmm/ichi.hmm
2 ni    hmm/ni.hmm
3 saN   hmm/saN.hmm
4 yoN   hmm/yoN.hmm
5 go    hmm/go.hmm
6 roku  hmm/roku.hmm
7 nana  hmm/nana.hmm
8 hachi hmm/hachi.hmm
9 kyuu  hmm/kyuu.hmm
```

空行はダメ。最後の行末には必ず改行。

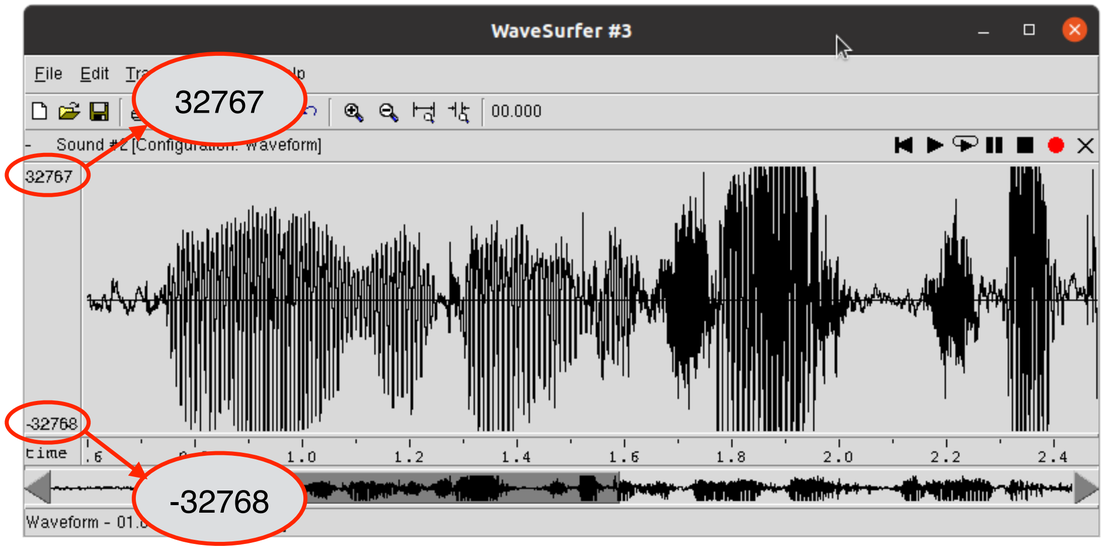
~/asr/wrecog/lib/wordlist単語名の一覧表

```
rei
ichi
ni
saN
yoN
go
roku
nana
hachi
kyuu
```

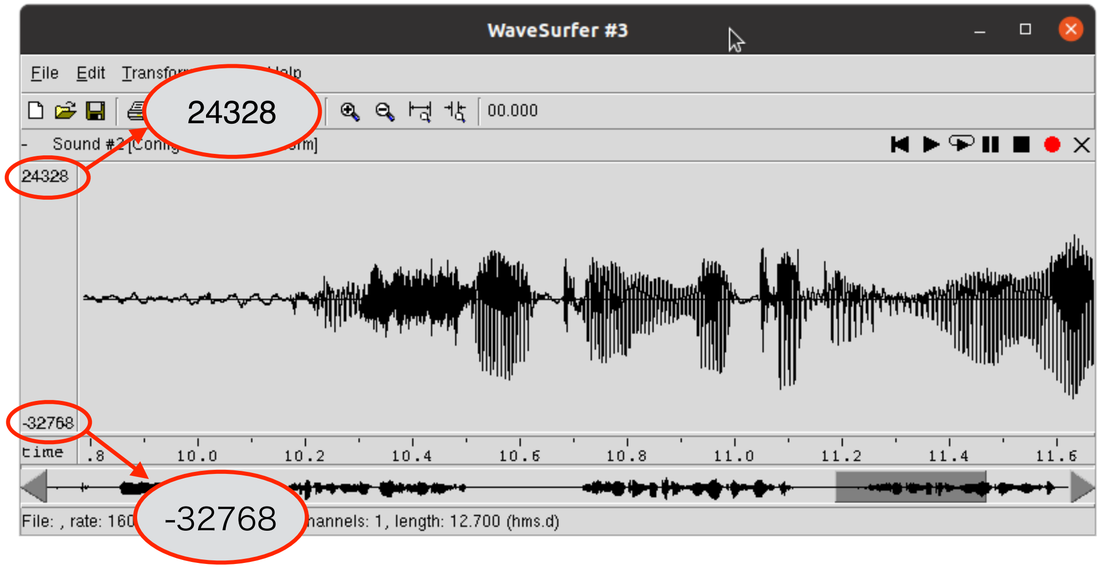
~/asr/wrecog/drawLC

単語HMMの学習曲線描画用
gnuplotスクリプト

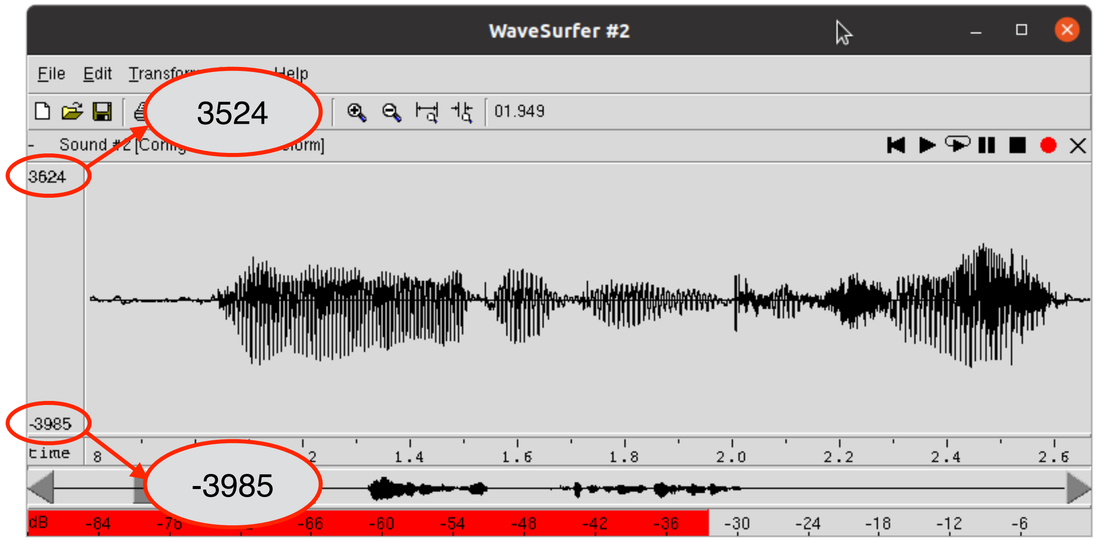
音声入力の適切な音量…過大になりがち。過小もダメ。



振幅過大で値域を超えた信号は正しく処理できない



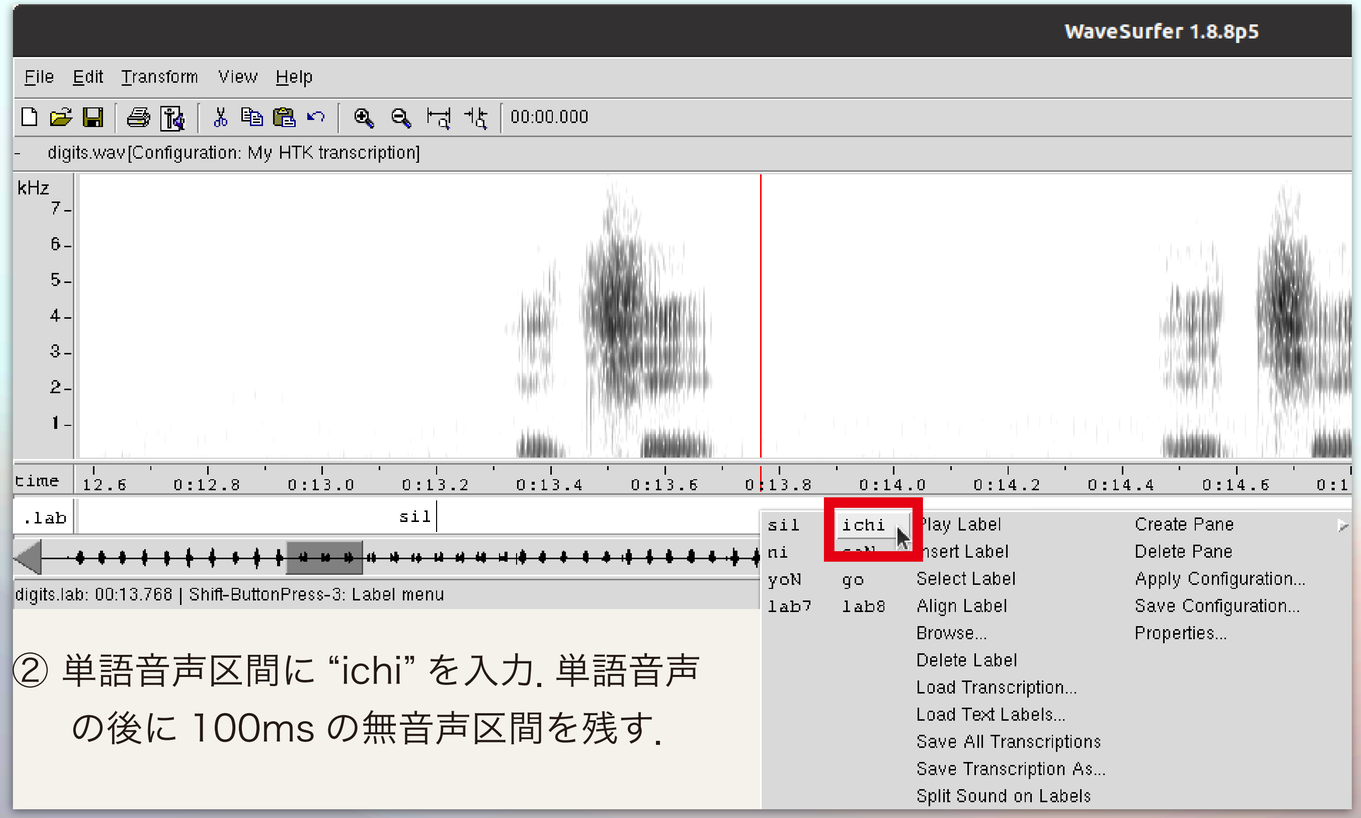
音量が適切ならば振幅値域内で
大きなダイナミックレンジを確保できる



振幅過小では波形情報を十分に活かし切れずSN比が悪くなりやすい（量子化誤差が大）

単語ラベルを正しく作る

- ラベルの設定
- 最初はsil
- 無音区間（sil）と単語は交互
  学習データに使わない区間はsilにする



② 単語音声区間に “ichi” を入力. 単語音声の後に 100ms の無音声区間を残す.

図6.10

- ラベルデータ要確認
  - ラベルファイル **`*.lab`** と音声波形ファイル **`*.wav`** は **`~/asr/wrecog/wav`** に保存
  - ラベル（テキストファイル）の内容を確認

```
[~/asr/wrecog]% ls -l wav
合計 3624
-rw-r--r-x 1 ta103027 faculty    5258 11月  15 11:02 digits.lab
-rw-r--r-x 1 ta103027 faculty 3701160 11月  15 11:06 digits.wav
[~/asr/wrecog]% head wav/digits.lab
0.0000000 1.6850000 sil
1.6850000 2.3141252 rei
2.3141252 2.8416402 sil
2.8416402 3.3983074 rei
3.3983074 4.0300000 sil
4.0300000 4.4375000 rei
4.4375000 5.2000000 sil
5.2000000 5.5600000 rei
5.5600000 6.3300000 sil
6.3300000 6.6175000 rei
[~/asr/wrecog]% tail wav/digits.lab
108.2125000 108.9925000 sil
108.9925000 109.3375000 kyuu
109.3375000 110.1050000 sil
110.1050000 110.4875000 kyuu
110.4875000 111.1850000 sil
111.1850000 111.5775000 kyuu
111.5775000 112.3250000 sil
112.3250000 112.6700000 kyuu
112.6700000 113.3475000 sil
113.3475000 113.6725000 kyuu
[~/asr/wrecog]%
```

図6.13

§6.5.1、§6.5.2

認識用設定ファイルの作成
単語音声学習データの作成
（ 録音→ラベル作成→切り出し ）

§6.5.3

スペクトル分析

§6.5.4

単語 HMM の学習

§6.5.5

学習の検証

§6.5.6

On-The-Fly 単語認識

単語HMM学習の要点

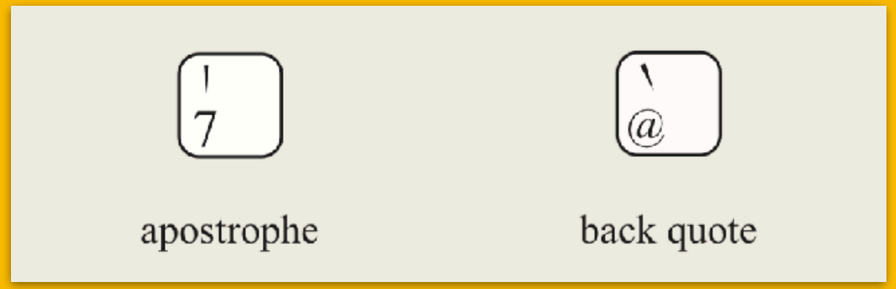
作業手順の補足：バッククオート（back quote）

- 「作成した単語音声の検聴」テキスト6-20頁
- 「スペクトル分析」テキスト6-20–21頁

```
foreach f (`ls wav/rei_??.wav`)
```

foreach 文の括弧の中の｀は back quote。

このテキストの PDF 版からコピー&ペーストで端末にコマンドを入力すると，表示フォントの影響でバッククオートが正しく入力されない。キーボードで。



§6.5.1、§6.5.2

認識用設定ファイルの作成
単語音声学習データの作成
（ 録音→ラベル作成→切り出し ）

§6.5.3

スペクトル分析

§6.5.4

単語 HMM の学習

§6.5.5

学習の検証

§6.5.6

On-The-Fly 単語認識

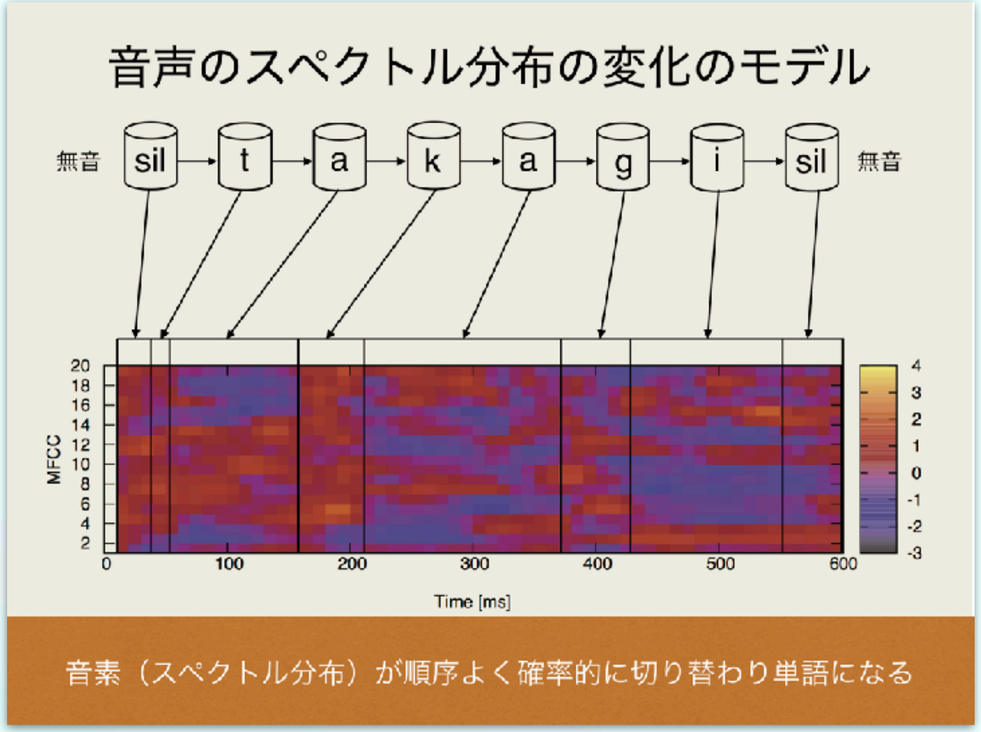
単語HMM学習の要点

HMMの状態数は ”目安” を参考に実験的に変えてOK。
学習に失敗するときは状態数を増減する：目安が７を超える場合は**減**らした方が良い

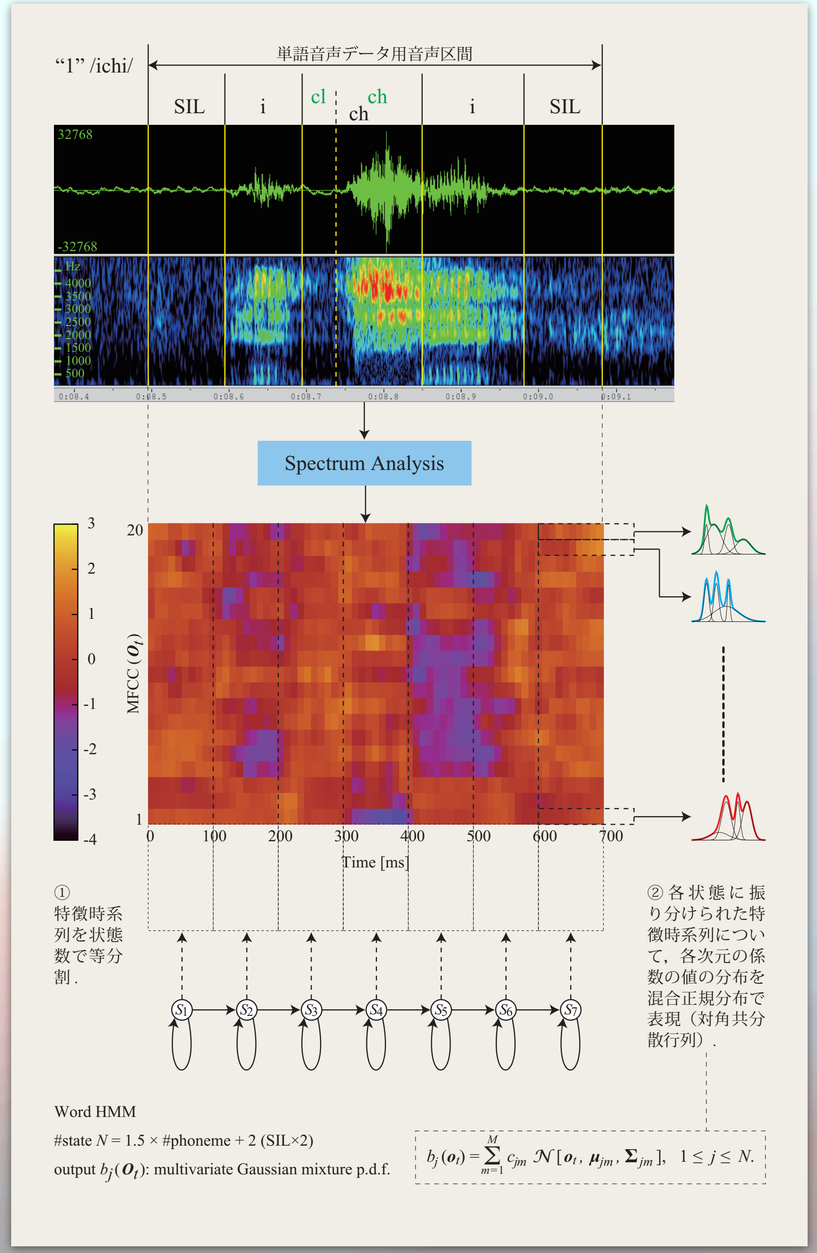
単語（正書法）	音素＋無音声区間	HMM 状態数（本実験での目安）
太郎	sil t a r oo sil	6
は	sil w a sil	4
学校	sil g a Q k oo sil	7
へ	sil e sil	3
行った	sil i Q t a sil	6

図 6.15: 本実験の条件において単語 HMM に割り当てる状態数の目安．音素 1 つあたり 1 状態．長音（/aa/，/ii/，/uu/，/ee/，/oo/），促音（/Q/）は 1 音素とみなす．単語の開始前と終了後に無音声（sil）のための 1 状態を割り当てる．

HMMの状態数は ”目安” を参考に適当に変えて良い：変えるなら少ない方が良い



第2日の説明資料



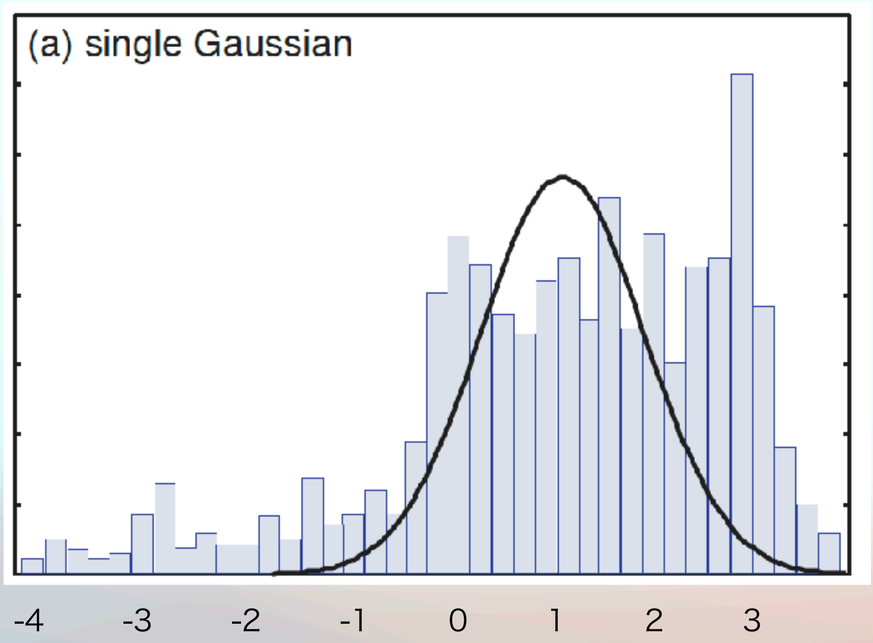
- 単語HMM学習プログラムtrainは、
  学習データのMFCC時系列を状態数で等分割する。
- 音響性質に応じた最適な分割をしているわけではない。
- 実用的にはこれで十分。

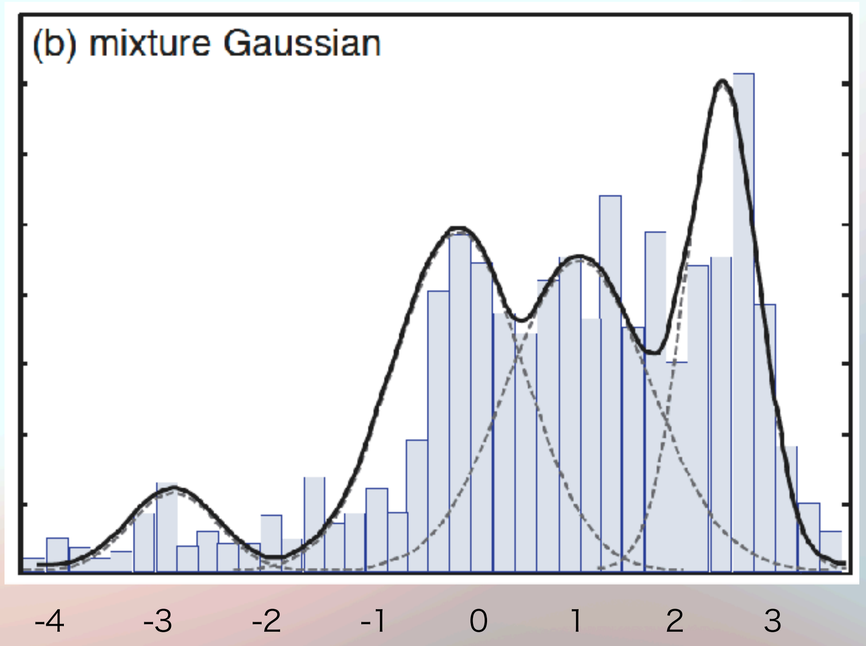
図6.2

数理実験と単語音声の違い

音声特徴量MFCCの確率密度関数

$$o_{t,i} \triangleq c_{t,i}$$ 第tフレームのMFCC第i係数

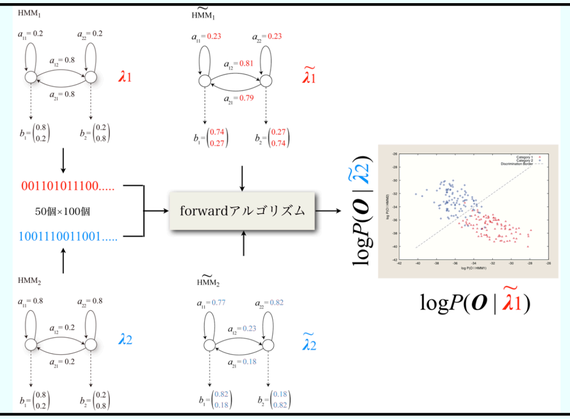
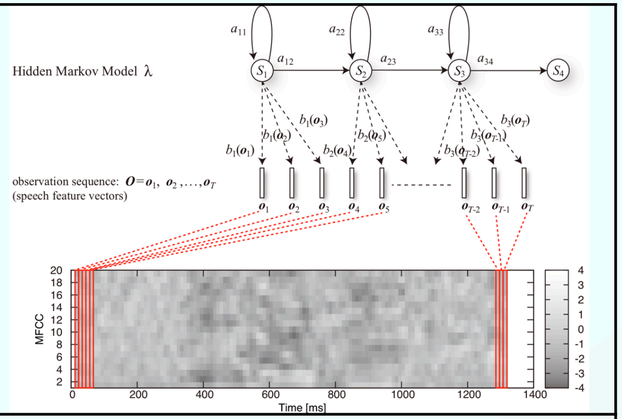
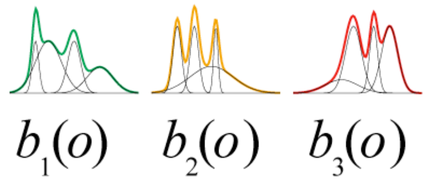




特徴量は多次元実数値ベクトル、複雑な分布を呈している。

多次元混合正規確率密度関数関数でモデル化している（テキスト5.4.2節）。

数理実験と単語音声の違い：音声特徴量MFCCの確率密度関数

		
出力シンボル	0, 1	R^{20}
出力確率	$$\boldsymbol{B}_1 = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}$$	gpdf.c $$b_j(\boldsymbol{o}_t) = \sum_{m=1}^{M} c_{jm}\ \mathcal{N}[\boldsymbol{o}_t, \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}]$$ 

Q: なぜフィルタバンク出力を離散コサイン変換してMFCCにする？
A: MFCCにすると次元間の相関が無くなりスペクトル分布のモデル化が容易になる

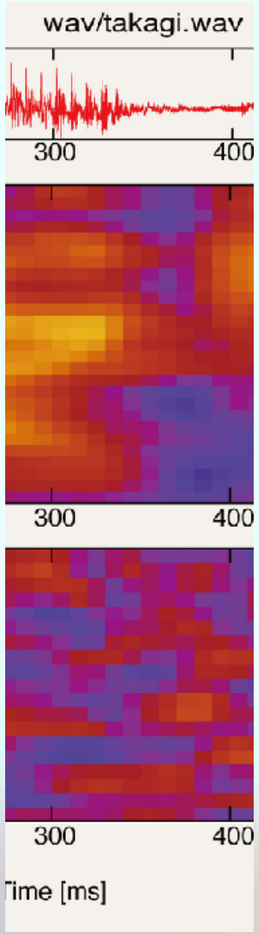
フィルタバンク出力

↓

離散コサイン変換

↓

メル周波数ケプストラム係数
MFCC



特徴量なので，多次元ガウス分布の確率密度関数を使います．

$$\mathcal{N}(\boldsymbol{o}_t, \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}) = \frac{1}{\sqrt{(2\pi)^D |\boldsymbol{\Sigma}_m|}} \exp\left\{-\frac{1}{2}(\boldsymbol{o}_t - \boldsymbol{\mu}_m)' \boldsymbol{\Sigma}_m^{-1} (\boldsymbol{o}_t - \boldsymbol{\mu}_m)\right\} \quad (5.33)$$

であり，D は $\boldsymbol{o}_t$ の次数，$'$ は転置，M は混合数，$\boldsymbol{\Sigma}_m^{-1}$ は $\boldsymbol{\Sigma}_m$ の逆行列，$|\boldsymbol{\Sigma}_m|$ は $\boldsymbol{\Sigma}_m$ の行列式を表わします．多次元データを利用した場合，$|\boldsymbol{\Sigma}_m|$ は共分散行列となりますが，この実験では，次元間の相関が無いと仮定した対角共分散行列を使用します．この場合，共分散行列は

$$\boldsymbol{\Sigma}_m = \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ 0 & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \sigma_D^2 \end{pmatrix} \quad (5.34)$$

となり，密度関数は，

$$\mathcal{N}(\boldsymbol{o}_t, \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}) = \frac{1}{\sqrt{\prod_{d=1}^{D} 2\pi\sigma_d^2}} \exp\left\{-\frac{1}{2}\sum_{d=1}^{D} \frac{(o_t^{(d)} - \mu_m^{(d)})^2}{\sigma_d^2}\right\} \quad (5.35)$$

で与えられます．ここで，$o_t^{(d)}$ は観測ベクトル $\boldsymbol{o}_t$ の第 d 次元，$\mu_m^{(d)}$ は第 m 混合の平均値ベクトル $\boldsymbol{\mu}_m$ の第 d 次元です．

相関が有ると仮定すると非対角成分を推定しなければならず、膨大な学習データが必要となる。実用上は対角共分散で十分。

On-The-Fly単語認識はシェルスクリプトで実装

この実習の実験はtcshで実行
コマンドラインで「tcsh」と入力するとシェルがtcshに切り替わる

§6.5.1、§6.5.2
認識用設定ファイルの作成
単語音声学習データの作成
（ 録音→ラベル作成→切り出し ）

§6.5.3
スペクトル分析

§6.5.4
単語 HMM の学習

§6.5.5
学習の検証

§6.5.6
On-The-Fly 単語認識

shell script
コマンド実行、ファイル操作、入出力処理などの手順を記述し、
一連の定型処理として実行するための仕組み。

- シェルスクリプトで実装しています。
- 音声入力 → 単語区間検出 → MFCC分析 → 単語認識 → 入力終了
- audioIN → vad → mfcc →_recog → hupAudioIN

音声区間検出の閾値

オンライン単語認識の
音声区間検出処理では
単語の前後に
最大100msの
無音声区間が残る

検出用閾値θの設定と
マイク入力音量が重要 … vuコマンド

```
[~/asr/wrecog]% vu

 Short Time Speech Power
-60     -50       -40      -30      -20      -10       0 dB
 +-------+-------+-------+-------+-------+-------+
rec WARN alsa: can't encode 0-bit Unknown or not applicable
.============================                *
```

テキストpp.6-25–26

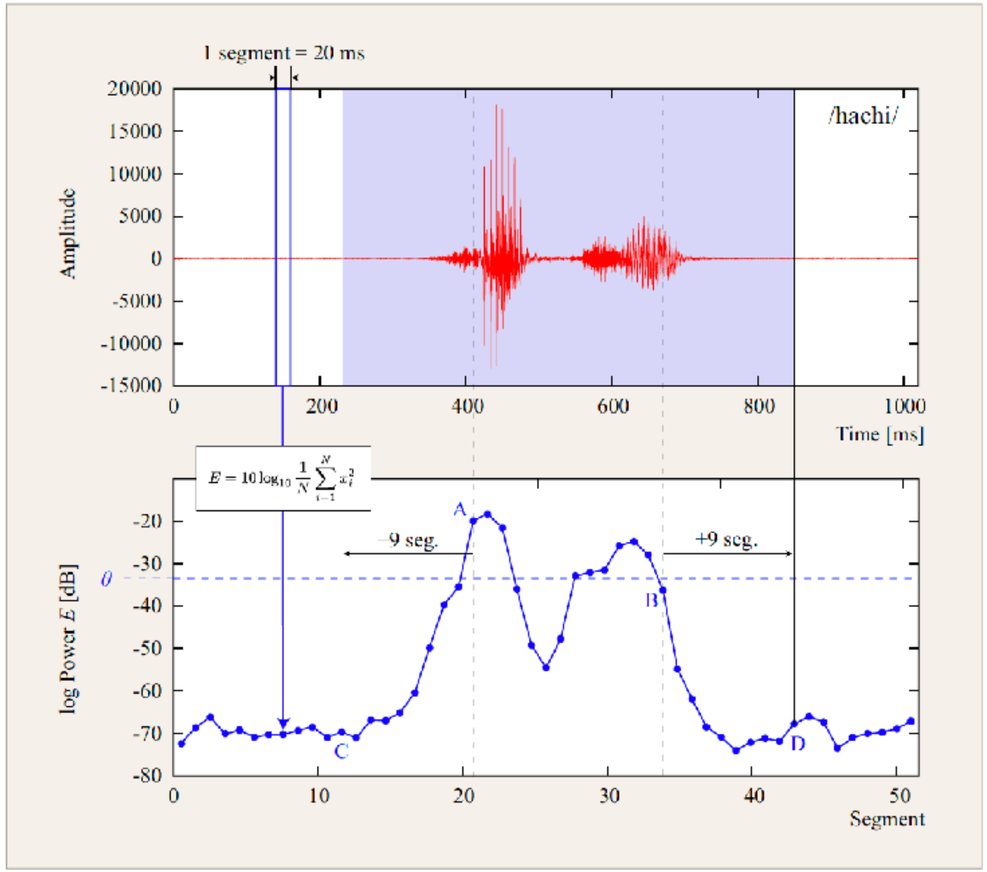


図 6.4: 音声区間検出の方式．上は音声波形（/hachi/）．下は対数パワーの変化グラフ．音声波形を 20 ms 幅のセグメント（区間）に区切り，セグメントの対数パワー $E = 10\log_{10}\frac{1}{N}\sum_{i=1}^{N}x_i^2$ を計算します．対数パワーの値が閾値 θ を初めて超えたセグメント A を検出します．A の後に閾値を初めて連続 5 セグメント下回った時の最初のセグメントを B とします．A の前 9 個目のセグメント C，および B の後 9 個目のセグメント D をそれぞれ音声区間の開始セグメント，終了セグメントとします．

音声区間検出プログラム ~/asr/sound/vad.c

~/asr/wrecog/program/recog　　シェルスクリプト

```
#!/bin/tcsh
  (中略)

#-------------------------------------
# オンライン認識
#-------------------------------------
while(1)
 echo -n "Returnキーを押してください"
 set buf = $<
 audioIN |\
 vad -p $th |\
 mfcc |\
 (_recog $hmmList && hupAudioIN)
end
```

| : pipe
前のコマンドの標準出力を後のコマンドの標準入力に流し込む

音声入力 ~/asr/sound/audioIN
`rec -q -r 16000 -c 1 -t raw -`

音声区間検出 ~/asr/sound/vad.c
テキスト「6.3.2音声区間の検出」

MFCC分析 ~/asr/wrecog/program/mfcc.c
stdin：音声波形、stdout：MFCC

audioINにhupシグナルを送る
~/asr/sound/hupAudioIN

単語音声認識 ~/asr/wrecog/program/_recog.c
stdin：MFCC、stdout：認識結果

cmdA && cmdB
cmdAが正常に終了したらcmdBを実行

On-The-Fly単語認識実行例

```
[~/asr/wrecog]% recog -25 lib/HMMList
Return キーを押してください

Isolated Word Recogition On-The-Fly
rec WARN alsa: can't encode 0-bit Unknown or not applicable vad_file= vadwav/76302.wav

--------------------
rank word (kana)= log-likelihood
--------------------
1. 8 (hachi)= -952.510926
2. 1 (ichi)= -977.724834
3. 3 (saN)= -1032.404311
4. 0 (rei)= -1087.581653
5. 2 (ni)= -1167.379263
6. 4 (yoN)= -1227.270401
7. 6 (roku)= -1240.591021
8. 7 (nana)= -1261.096937
9. 5 (go)= -1360.261797
10. 9 (kyuu)= -1373.558098
ハングアップ

Return キーを押してください
```

音声認識起動コマンド「-25」はvuコマンドで決める

音声区間検出で切り出された音声波形

第1位の認識候補は「８」で、尤度 $\log P(\boldsymbol{O}|\lambda)$ は -952.510926

On-The-Fly単語認識の評価？？？

- 受講者の自主性と工夫に期待します。
  - 認識対象の単語を複数回入力して正解数を数える
  - 発音の仕方を変える
  - 認識対象以外の単語を入力する（似た発音／全く違う発音の単語）
  - 別の人に使ってもらう（学習話者と認識話者が異なる条件）
  - その他

実際に認識に用いられた音声波形を見ると分かることも多い

- ★ 音声区間検出によって切り出されて尤度計算に用いた音声波形は vadwav/*.wav (ファイル名は認識結果の上に表示されている）に保存されています。
- ★ WaveSurferで開いて波形を観察したり聴いたりしてみる。
- ★ ありがちな失敗：音が割れている、入力単語が途中で切れている（語頭または語尾）。

FAQ

Q：recogfコマンドを実行するとエラーが出る。

recogf: error reading word names, HMM Labels, and HMM parameter file names from lib/HMMList

A：lib/HMMListに空行があってはなりません。最後の行末には必ず改行を入れてください。そうなってない場合に上記のエラーが出ます。

Q：On-The-Fly単語認識で発声しても認識結果が表示されません。

A：音声区間検出が失敗しています。パワーの閾値を変更する、あるいはマイク入力の音量を増減してください。

第3回レポート締切は2024年12月18日

Q：On-The-Fly単語認識で ”**Broken Pipe**”と表示される。

A：recogコマンドが内部で呼び出しているコマンドのいずれかが実行できていません。コマンドがコンパイルされていない、あるいはコマンドパスの設定が適切でないことが原因。

【対策１】音声認識に必要なコマンドのコンパイルの確認
テキスト p.5-25

```
[~/asr/wrecog]% make -C program all
```

【対策2】「1.7.2 環境設定の追加」確認。tcshで実施。

【対策3】recogコマンドを編集し、実習用コマンドをフルパス指定する。「6.5.7 プログラム構成」参照。