

卒業論文
SVM を用いた楽器音の識別

5年情報工学科
東海林研究室
出席番号10番
杉山幸恵

目次	ページ数
1.目的	3
2.研究概要	4
3.特徴抽出	5
3-1.FFT	5
3-2.ケプストラム	5
4.識別器作成	6
4-1.SVM	6
4-2.最大マージン SVM	6
4-3.カーネル関数	7
4-4.マージン	8
4-5.1-ノルムソフトマージン最適化	8
4-6.学習率を使った実際の超平面の求め方	8
5.認識, 識別	9
6.実験	10
6-1.使用データ	10
6-2.特徴抽出	10
6-3.識別器作成	12
6-4.認識率, 正解率の比較	12
7.プログラムリスト	13
8.実験結果	19
8-1.分散 σ 二乗毎の認識率の比較	19
8-2.分散 σ 二乗毎の正解率の比較	22
8-3.各識別器の認識率と正解率の比較	24
9.まとめ	25
10.参考文献	26

1.目的

CDを聞いて、その曲の楽譜が欲しいと思ったとしても、全ての楽譜が一般に売られている訳ではないので、手に入れることが難しい場合がある。その時、自分の耳で聞いて採譜をすることも、普通の聴覚を持った人にとっては困難である。オーケストラや吹奏楽のように、様々な楽器で構成されている曲の採譜をするのはさらに困難であり、何種類の楽器で構成されているのか聞き取ることさえできない場合が多い。そのため、近年、自動採譜の研究が盛んに行われている。その際、音を楽器毎に分けて採譜を容易にするために、入力音が何の楽器の音であるかを識別する識別器が必要である。

従来、識別器としては、テンプレートマッチング、ニューラルネットワーク、GMM、EM アルゴリズム、ベイジアンネットワークなどが用いられ、新しい手法としてサポートベクターマシンによるものが増えている。識別器作成には入力音の特徴量が使用され、特徴抽出手法としては、FFT、MFCCなどが用いられている[1][2]。しかし、MFCCは計算量が多いため、本論文では特徴抽出手法にFFTとケプストラムを使用することを提案し、識別器にサポートベクターマシンを使用することを提案する。

よって、本研究の目的は入力音が何の楽器の音であるかを識別する識別器を作成することである。また、その識別器に実際にデータを入力して認識率と正解率を求め、特徴量の違いによる識別器の認識率と正解率の比較を行う。

2.研究概要

初めに,各楽器の特徴量をFFT(高速フーリエ変換),ケプストラムを用いて抽出する.次に,それぞれの特徴量毎に各楽器の識別器をSVM(Support Vector Machine)を用いて作成し,入力音の演奏楽器を識別する(図1).その後,FFTによる識別器とケプストラムによる識別器の認識率と正解率の比較を行う.

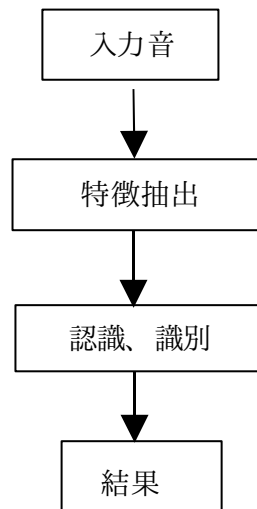


図1.楽器音認識の流れ

3.特徴抽出

今回は FFT とケプストラムを用いて特徴抽出を行う。

3-1.FFT

FFT により入力波形のスペクトルを得ることができる。楽器音のスペクトルの場合、楽器情報と音高情報が含まれ、それらの特徴は振幅スペクトルまたはその 2 乗値であるパワースペクトルに現れる。今回はパワースペクトルを使用する。パワースペクトルの計算式は、

$$C_k^2 = a_k^2 + b_k^2$$

となる。この時、

ak: スペクトルの実数部

bk: スペクトルの虚数部

である[3][4]。

3-2.ケプストラム

ケプストラムは、フーリエ変換によって求めたパワースペクトルを対数に変換し、さらに逆フーリエ変換を行って得られる。この処理を行うことで楽器情報と音高情報を分離させることができ、低次には楽器情報が、高次には音高情報が現れる。

上記の計算式でパワースペクトルを求め、常用対数をとる。逆フーリエ変換する際、実数部と虚数部が必要なため、実数部には対数をとったパワースペクトルを、虚数部には 0 を入れて逆フーリエ変換を行う。結果は実数部のみを用いる[5][6][7]。

フーリエ変換と逆フーリエ変換の違いは偏角の符号と計算結果を N で割るかどうかの違いだけなので、フラグを立てることで同じプログラムで両方行うことができる。

4.識別器作成

各楽器の特徴量を抽出し,それを SVM に学習させ,ある楽器をクラス 1,それ以外の楽器をクラス-1 に対応させることで識別器を作成する。

4-1.SVM

識別器作成に用いた SVM とは,パターン認識の学習モデルの 1 つであり,2 クラスの識別器を構成する手法である[8][9].与えられた学習データの中に含まれる,サポートベクトルと呼ばれるクラスの境界面近くに位置するベクトルとクラスを 2 つに分離する平面である分離超平面との最小距離をマージンという.線形で分類が難しい際には,カーネルによって入力空間をより高次の特徴空間に写像し,そこで線形分離を行うことで非線形の問題にも適用が可能になる.従来,パターン認識の分野でよく用いられていた逆誤差伝播法によって学習を行う多層パーセプトロンに比べ,

- ・ 高い汎化性能(学習に用いていない未知のデータに対して識別を行った際の性能)を持っている
- ・ 2次最適化問題として定式化されるため学習の結果,最適解を得る事ができる

という特徴を持っている。

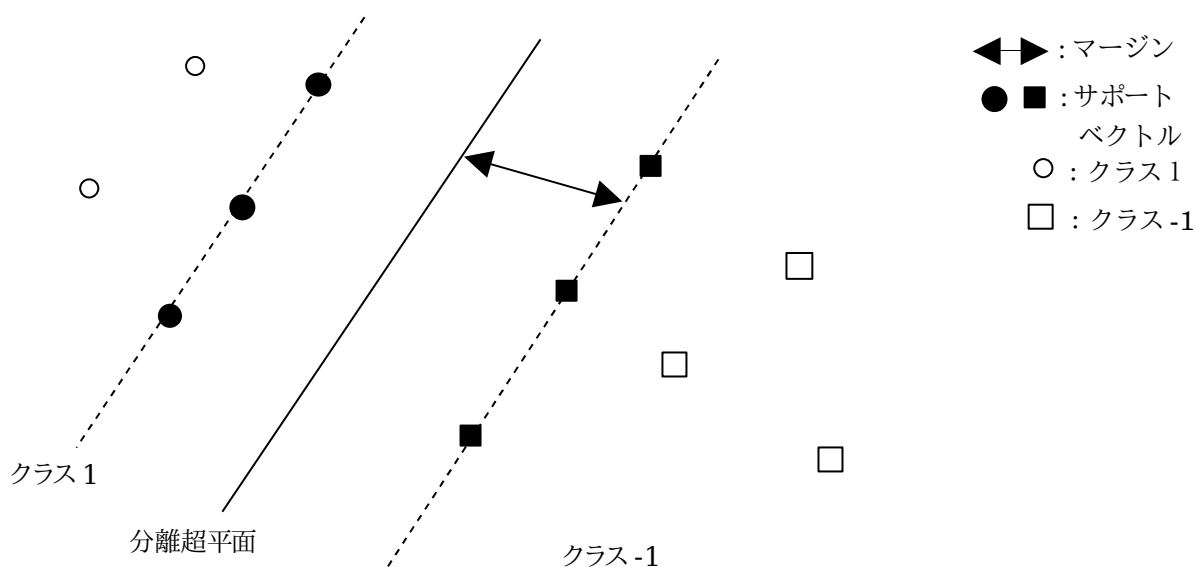
SVM でのクラス分類の目的は,高次元特徴空間でうまく分離する超平面を,計算量的に効率良く学習する方法として提供することである。

4-2.最大マージン SVM

最大マージン SVM とは,マージンを最大とする超平面を求める方法であり,SVM の最も単純なモデルである(図 2).分離超平面の式は,

$$f = \langle \mathbf{w} \cdot \mathbf{x} \rangle + b \cdots (1) \quad (\mathbf{w} : \text{重みベクトル}, \mathbf{x} : \text{入力ベクトル})$$

となる.この式に値を入れ+ならクラス 1,-ならクラス-1 に識別する。



4-3.カーネル関数

(1)を変形すると,

$$\sum w_i x_i + b = 0 \cdots (2)$$

となる. w と b を求める方法を主形式という.ここで,

$$\mathbf{W} = \sum \alpha_i y_i \mathbf{x}_i \cdots (3)$$

とおく.この式を主形式に代入すると,

$$\sum \alpha_i y_i \langle \mathbf{x}_i \cdot \mathbf{x} \rangle + b \cdots (4)$$

となる.この式から α と b を求める方法を双対形式という. $\langle \mathbf{x}_i \cdot \mathbf{x} \rangle$ は内積であり,これを $k(x,z)$ に置き換えると,

$$\sum \alpha_i y_i k(x,z) + b \cdots (5)$$

となる. $K(x,z)$ をカーネル関数と呼び,ベクトルの内積を一般化した関数である[10].

入力関数 x を非線形写像 Φ を使って高次元特徴空間にすると平面で分離させることができる(図3).

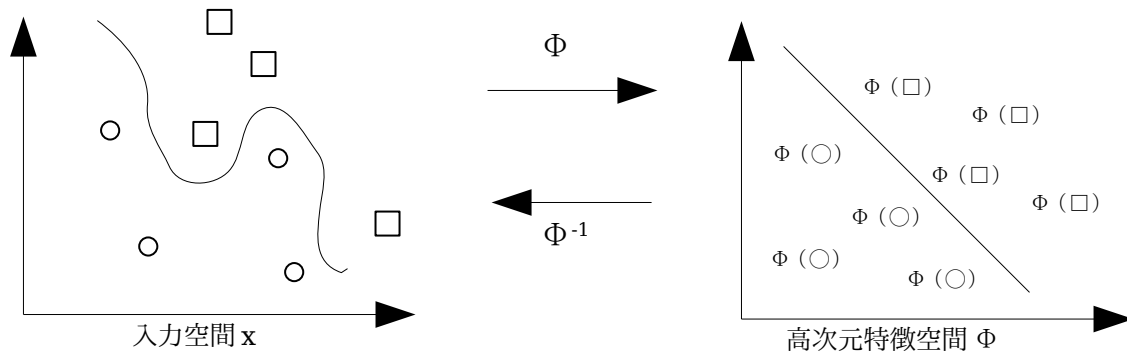


図3. Φ による高次元化

Φ によって次元を増やしたことで高次元特徴空間での主形式は,

$$f(x) = \sum w_i \Phi_i(x) + b \cdots (6)$$

となり,双対形式に変形すると,

$$f(x) = \sum \alpha_i y_i \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) \rangle + b \cdots (7)$$

となる. $\Phi(x)$ は無次元空間への写影なので直接決めるのは不可能であるので,そこで使用するのがカーネル関数である.カーネル関数 K で表すと(7)は(5)のようになる.

ここで K がカーネルとなる条件は以下の3つである.

- ①対称行列
- ②三角不等式
- ③マーセル条件(行列が半正定値,全ての固有値が0以上)

この3つが成り立たなければカーネルとなることはできず,成り立たない場合はノルム² < 0となる点がある.

カーネルには多項式カーネルとガウスカーネルなどがあり,今回はガウスカーネルを使用する.ガウスカーネルの式は以下の通りである.

$$K(x,z) = \exp(-\|x-z\|^2 / \sigma^2) \cdots (8)$$

4-4. マージン

マージンには 2 種類のマージンがある.

・関数マージン $\gamma_i = y_i(w \cdot x + b) \cdots (9)$

・幾何マージン $\gamma_i = y_i(w \cdot x / |w| + b / |w|) \cdots (10)$

今回は関数マージンを使用する. 関数マージンは w のノルムを 1 に正規化したマージンである.

4-5.1-ノルムソフトマージン最適化

最大マージン SVM では, 特徴空間の次元数が学習データの数よりも大きければ完全に識別できると仮定されているが, 一般にはそうでない場合も考えられる. その場合, 最適化を行うことができない. この時, $C > \alpha_i \geq 0$ のように制限をつけることで対処することができる. これをソフトマージンという. 最大マージンは重要な概念であるが, そのまま実世界問題に用いることができない. なので 1-ノルムソフトマージン最適化を行って識別器を作成する.

$K(x, z)$ において, 特徴空間で陰に線形分離可能なデータ

$$S = ((x_1, y_1), \dots, (x_l, y_l)) \cdots (11) \quad (x: \text{座標}, y: 1 \text{ 又は } -1)$$

を考え, パラメータ α^* と b^* は 2 次最適化問題

$$W(\alpha) = \sum \alpha_i - 1/2 \sum y_i y_j \alpha_i \alpha_j K(x_i, x_j) \cdots (12) \rightarrow \text{最大化}$$

の最適解と仮定する. ただし,

$$\sum y_i \alpha_i = 0 \cdots (13)$$

$$0 \leq \alpha_i < C \cdots (14) \quad (i=1, \dots, l, C < \infty)$$

を条件とする. すると, 決定規則

$$\text{sgn}(f(x)) \cdots (15) \quad \text{sgn}(x) = \begin{cases} 1 & (x \geq 0) \\ -1 & (x < 0) \end{cases}$$

$$(\text{ただし}, f(x) = \sum y_i \alpha_i^* K(x_i, x) + b^* \text{ とする})$$

は, 陰に $K(x, z)$ により定義される特徴空間の最大マージン超平面に等しく, その超平面は幾何マージン

$$\gamma = (\sum \alpha_i^*)^{-1/2} \cdots (16)$$

を持つ. 関数マージンは

$$y_j \cdot f(x_j) = y_j \{ \sum y_i \alpha_i^* K(x_i, x_j) + b^* \} \cdots (17)$$

であり,

$$\text{幾何マージン} = \text{関数マージン} / |w| \quad (\gamma = (\sum \alpha_i^*)^{-1/2} = 1 / |w|)$$

であるので, 関数マージンは 1 となる. しかし, 関数マージン 1 の平面は ∞ にあるので, その中でも $|w|$ を最小化する平面を分離超平面とする.

4-6. 学習率を用いた実際の超平面の求め方

b を固定値として設定すると, 最適化問題の双対形式は,

$$W(\alpha) = \sum \alpha_i - 1/2 \sum y_i y_j \alpha_i \alpha_j K(x_i, x_j) \cdots (18) \rightarrow \text{最大化}$$

(ただし, $0 \leq \alpha_i < C, i=1, \dots, l$)

となる. 更新規則

$$\alpha_i \leftarrow \alpha_i + \eta \frac{\partial W(\alpha)}{\partial \alpha_i} \cdots (19)$$

$$(\frac{\partial W(\alpha)}{\partial \alpha_i} = 1 - y_i \sum \alpha_j y_j K(x_i, x_j))$$

を用いて, 適当な学習率 η によりループをする. すべての α_i を正の象限に同時に保てば, $W(\alpha)$ を最大化できることになる. α_i が負になった時は, $\alpha_i \leftarrow 0$ とすることで正の象限に強要できる. 同様に, ソフトマージンによる α_i の上限は乗数が C より大きくなれば, $\alpha_i \leftarrow C$ と設定することで強要できる.

こうして, f を求めるのに必要な α_i を求めることができる.

5. 認識, 識別

未知音の特徴量を FFT またはケプストラムで抽出し, その特徴量を識別器に通して認識を行い, クラスを識別する. 識別器に通すということは, 分離超平面 f にデータを入力するということである. この時の f 符号でクラスの識別をする. ある楽器をクラス 1, それ以外の楽器をクラス -1 に対応させ, f の値が + であったらクラス 1, - であったらクラス -1 であると識別できる. また, f の絶対値 $|f|$ が 1 より大きい場合は確実に識別されていることを表し, 0 に近い場合は識別が困難であることを表す.

6.実験

ピアノとギターの2種類の識別を行う。クラス1をピアノに、クラス-1をギターに対応させる。

6-1.使用データ

1音ずつ録音し,wavファイルとする。学習データとして音高の異なるピアノ音とギター音を各20音,テストデータとして音高の異なるピアノ音とギター音を各10音をサンプリング周波数44100Hz,モノラル16bitで録音する。

6-2.特徴抽出

wavファイルを読み込み,特徴抽出前にデータに処理を施す。入力した始めの部分は音が安定せず特徴抽出に向かないため,先頭から0.45秒カットした後の定常部分を入力データとする。サンプリング周波数44100Hzであるので,1秒間に44100点サンプリングする。よって,0.45秒にあたる20000点目からデータを使うことで,先頭から0.45秒をカットすることができる。次に,このデータの正規化を行う。入力データは,20000点目から窓幅 $N=512$ 点を加算した20512点目までを使用する。このデータの一番大きい振幅 \max を求め,データを \max で割ることで正規化する。ただし,FFTの場合 \max で割ったデータを用いてFFTすると多くのデータで高周波部分が0となるため,ピアノの場合は100倍,ギターの場合は10000倍してから \max で割る。そうすることで,高周波部分の特徴量を用いることができる。最後に,切り出した信号の最初と最後に不連続性が生じるとスペクトルに影響を与えるので,切り出した信号の最初の値と最後の値を等しくするために窓関数をかける。今回は音に関して有効とされるハニング窓を使用する(図4)。ハニング窓は,主な周波数成分から少し離れた,小さな成分を検出するのに適している。式は,

$$h[t]=0.5-0.5\cos(2\pi t/N)$$

である。

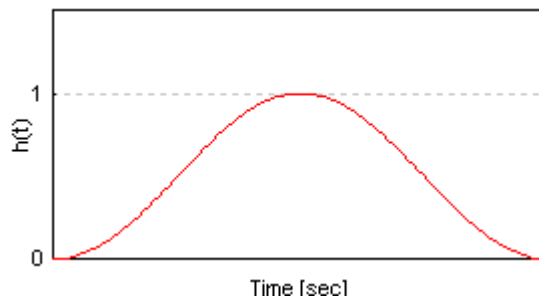


図4. ハニング窓

前処理後,特徴抽出処理を行う。窓幅 N は512点とする。

例として、ドの音を入力した時のパワースペクトルとケプストラムによる楽器毎の特徴の違いをグラフで表す。

・入力データ例 (C4)

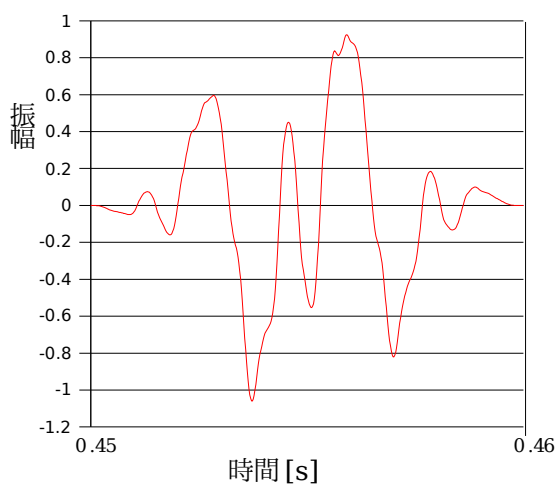


図5.ピアノ入力データ (C4)

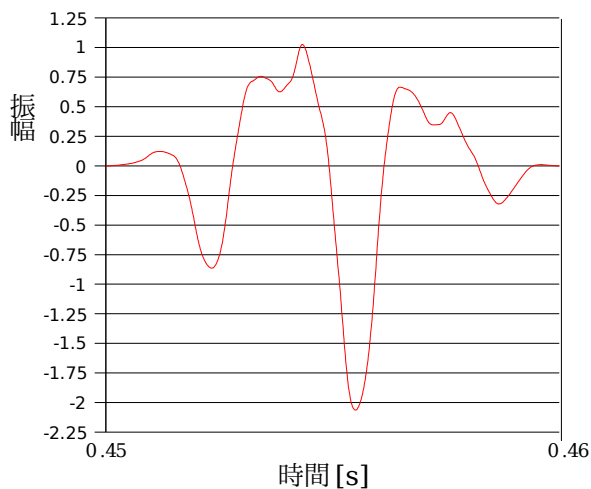


図6.ギター入力データ (C4)

まず入力データのグラフで違いを比較してみる.同じ高さのドの音でも,楽器によって波形が異なることがわかる.

・パワースペクトル

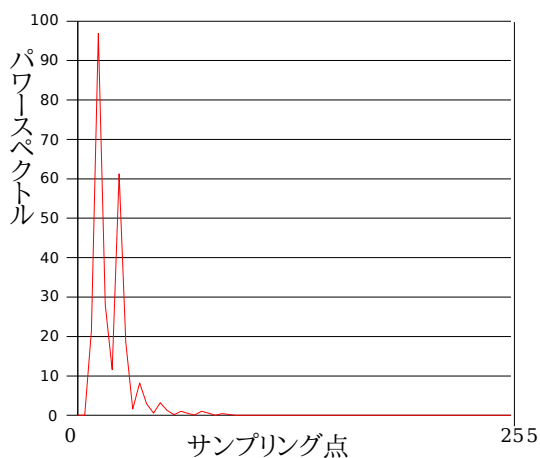


図7.ピアノのパワースペクトル (C4)

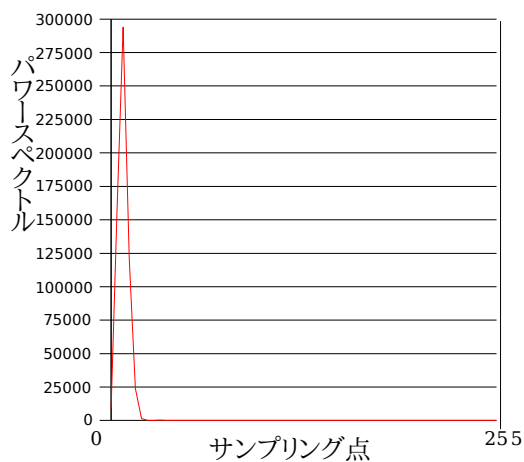


図8.ギターのパワースペクトル (C4)

パワースペクトルで表しても, ピアノとギターで波形が異なることがわかる. 0 や $N-1$ に近い部分には低周波成分が, $(N-1)/2$ に近い部分には高周波成分が現れる. パワースペクトルでは, 高周波成分がほとんどみれなくなっている[11].

・ケプストラム

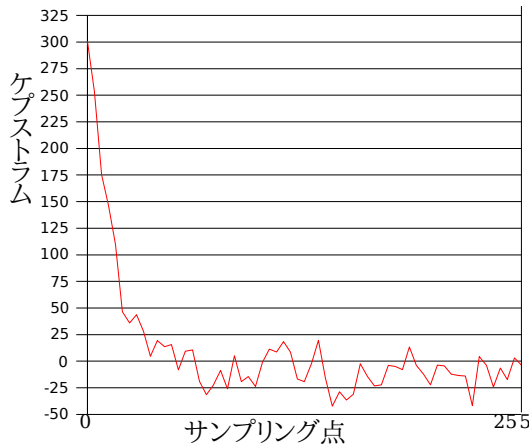


図 9.ピアノのケプストラム(C4)

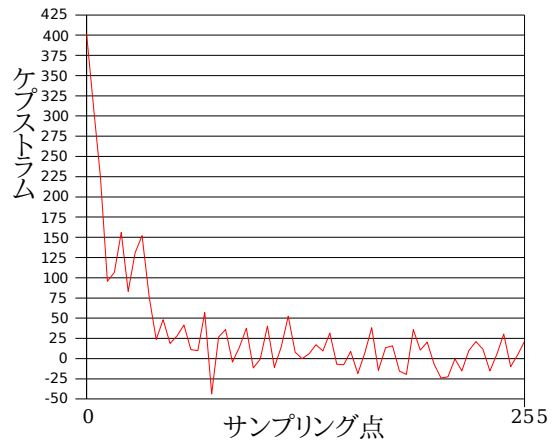


図 10.ギターのケプストラム(C4)

ケプストラムでも、ピアノとギターで異なっていることがわかる。この手法により、パワースペクトルでは見られなくなっていた高周波部分も見られるようになり、この部分の特徴も用いることができるようになる。

6-3.識別器作成

パワースペクトルの特徴量のみを用いたもの、ケプストラムの特徴量のみを用いたもの、およびパワースペクトルとケプストラムの両方の特徴量を用いたものの3つの識別器を作成する。パワースペクトル、ケプストラムは $(N-1)/2$ を中心に左右対称となっているため、識別器作成に使用するデータは256次元とする。

超平面 f を求める際、256次元のデータがクラス1、クラス-1のどちらに属しているかが必要なため、データ毎にピアノの場合は1、ギターの場合は-1を付加する。そのデータをSVMに学習させ、分離超平面 f を求めることで識別器作成となる。

カーネルを求める際に用いる分散 σ^2 の値が正しくない場合、学習データを入力した時の f の値が全て同じであったり、テストデータを入力しても f の値は0のまま識別できていないなど、識別器が作成できない原因となるので、 σ^2 を調整しながら最も良い識別を行う f を求めなければならない。SVMに関数マージンを使用しているため、学習データを入力した時は $|f|$ が1に近いもの程良い識別器である。テストデータを入力した時は f が1または-1を越えている場合は確実に識別されていることを表し、0に近い場合はクラスを判断し難いことを表すので、 f が1または-1を越えているもの程良い識別器である。

6-4.認識率,正解率の比較

認識率とは、SVMにより学習データがどの程度分離できたかの割合のことである。求めた分離超平面 f に学習データを入力した時、 f の符号と入力した学習データの符号がいくつ一致したかで求める。

正解率とは、SVMによりテストデータがどの程度識別できたかの割合のことである。テストデータとして、あらかじめわかっているピアノ音とギター音を10個ずつ用意し、分離超平面 f に入力した時、いくつ正解したかで求める。この時の正解とは、入力音がピアノ音なら f が+の符号、ギター音なら f が-の符号になることである。

7. プログラムリスト

・ 0.45 秒カット

```
j=20000; //入力データの先頭 0.5 秒カット
//一番大きい部分を探す
for(k=20000;k<(20000+N);k++) //N=512
{
    if(max<=abs(buf_in[k])) //buf_in[k]:wav ファイルから読み込んだデータ
    {
        max=buf_in[k];
    }
}
```

・ 正規化 (FFT の場合)

```
if(x<20){ //ピアノの学習データの正規化
    for(i=0;i<N;i++){
        input[i].Re=(double)buf_in[j]*100/max;
        input[i].Im=0.0;
        j++;
    }
}
else{
    if(x<40){ //ギター of 学習データの正規化
        for(i=0;i<N;i++){
            input[i].Re=(double)buf_in[j]*10000/max;
            input[i].Im=0.0;
            j++;
        }
    }
    else{
        if(x<50){ //ピアノ of テストデータの正規化
            for(i=0;i<N;i++){
                input[i].Re=(double)buf_in[j]*100/max;
                input[i].Im=0.0;
                j++;
            }
        }
        else{ //ギター of テストデータの正規化
            for(i=0;i<N;i++){
                input[i].Re=(double)buf_in[j]*10000/max;
                input[i].Im=0.0;
                j++;
            }
        }
    }
}
```

• 正規化(ケプストラムの場合)

```
for(i=0;i<N;i++){
    input[i].Re=(double)buf_in[j]/max;
    input[i].Im=0.0;

    j++;
}
```

• ハニング窓をかける

```
for(j=0;j<N;j++){
    w_input[j].Re=input[j].Re*(0.5-0.5*cos(2*PI*j/N));//ハニング窓
    w_input[j].Im=0.0;
}
```

• FFT/IFFT

```
//ビット逆転処理
void swap(NUMBER *w_input,int i,int j)
{
    double xa,ya;

    xa=w_input[i].Re;
    w_input[i].Re=w_input[j].Re;
    w_input[j].Re=xa;
    ya=w_input[i].Im;
    w_input[i].Im=w_input[j].Im;
    w_input[j].Im=ya;

}
```

```
void BIT_R(NUMBER *w_input,int n)
{
    int i,j,n2;

    j=0;
    for(i=0;i<=n-2;i++)
    {
        if(i<j) swap(w_input,i,j);
        n2=n/2;
        while(j>=n2){
            j=j-n2;
            n2=n2/2;
        }
        j+=n2;
    }
}
```

```

//バタフライ演算
void Butterfly(NUMBER *w_input,int i,int iw,double C,double S)
{
    double wr,wi;

    wr=(w_input[iw].Re*C)-(w_input[iw].Im*S);
    wi=(w_input[iw].Re*S)+(w_input[iw].Im*C);
    w_input[iw].Re=w_input[i].Re-wr;
    w_input[iw].Im=w_input[i].Im-wi;
    w_input[i].Re =w_input[i].Re+wr;
    w_input[i].Im =w_input[i].Im+wi;

}

//FFT 又は IFFT
void FFT_IFFT(NUMBER *w_input,int n,int flag)
{
    int i,j,L,m,iw,Lp,Lp2,n2;
    double D,ARG,C,S;

    BIT_R(w_input,n);

    m=0;
    n2=n;
    while(n2!=1)
    {m++;n2=n2/2;}
    for(L=1;L<=m;L++){
        Lp=(int)pow(2.0,(double)L);
        Lp2=Lp/2;
        D=-(double)flag*PI/Lp2;
        ARG=0.0;
        for(j=0;j<Lp2;j++){
            C=cos(ARG);
            S=sin(ARG);
            ARG+=D;
            for(i=j;i<n;i+=Lp){
                iw=i+Lp2;
                Butterfly(w_input,i,iw,C,S);
            }
        }
    }
    if(flag==fft){
        for(i=0;i<n;i++){
            w_input[i].Re=w_input[i].Re/(double)n;
            w_input[i].Im=w_input[i].Im/(double)n;
        }
    }
}

```

・パワースペクトル

```
for(j=0;j<N;j++){
    fft_output[j].a=0.0;
    fft_output[j].a=fft_output[j].Re*fft_output[j].Re+fft_output[j].Im*fft_output[j].Im;
    pawaspe_output[x][j]=fft_output[j].a; //pawaspe_output[x][j] :表示用変数
}
```

・ケプストラム

```
//パワースペクトルの対数をとる
for(j=0;j<N;j++){
    w_input[j].Re=log10(fft_output[j].a);
    w_input[j].Im=0.0;
}

//IFFTをする
flag=ifft; // fft : 1 , ifft : -1
FFT_IFFT(w_input,N,flag);
ifft_output=w_input;
for(j=0;j<N;j++)
{
    pawaspe_output[x][j]=ifft_output[j].Re;
}
```

・識別器作成

```
//超平面を求める
void chouheimen(double *x[],double a[])
{
    double *n,s;
    int i,j;
    double current_w=0,pre_w=0;

    double K(double *x,double *z); //カーネルの計算関数
    n = ( double* )malloc( sizeof(double)*N );
    for(i=1;i<=l;i++)
    {
        a[i]=0;
        n[i]=0.2/K(&x[i][0],&x[i][0]);
    }

    for(;;)
    {
        for(i=1;i<=l;i++)
        {
            s=0;
            for(j=1;j<=l;j++)
                s=s+a[j]*x[j][257]*K(&x[i][0],&x[j][0]);
            a[i]=a[i]+n[i]*(1-(x[i][257]*s));
        }
    }
}
```



```

        if(a[i]<0)a[i]=0;
        else if(a[i]>C)a[i]=C;
    }
    double t1=0;
    for(i=1;i<=l;i++)
        t1+=a[i];
    double t2=0;
    for(i=1;i<=l;i++)
    {
        for(j=1;j<=l;j++)
            t2+=x[i][257]*x[j][257]*a[i]*a[j]*K(&x[i][0],&x[j][0]);
    }
    current_w=t1-t2/2;
    if(fabs(current_w-pre_w)<TYNY_NUM)break;
    pre_w=current_w;
}
free(n);
}

```

//カーネルの計算

```

double K(double *x,double *z)
{
    double S=0;
    int k;
    for(k=1;k<=256;k++)
    {
        S+=(x[k]-z[k])*(x[k]-z[k]);
    }
    S=S/256.0;
    double A=-S/sigma2;
    return exp(A);
}

```

・認識率を求める

```

for(i=1;i<=l;i++)
{
    f=0;
    for(j=1;j<=l;j++)
    {
        f+=x[j][257]*a[j]*K(&x[j][0],&x[i][0]);
    }
    if(f>0) {
        if(x[i][257]==1.0){
            seikai+=1;
        }
    }
    else{
        if(x[i][257]==-1.0){

```

```

        seikai+=1;}
    }
    fprintf(s1,"f=%lf¥n",f);
}
}
seikai=seikai*100/l;
fprintf(s1,"認識率は%lf¥n",seikai);

```

• 正解率を求める

```

for(i=1;i<=m;i++)
{
    f=0;
    for(j=1;j<=l;j++)
    {
        f+=x[j][257]*a[j]*K(x[j],test[i]);
        printf("k=%lf¥n",K(x[j],test[i]));
    }
    if(f>0) {
        piano+=1;
    }
    else{
        guitar+=1;
    }
    fprintf(s1,"f=%lf¥n",f);
    printf("f=%lf¥n",f);
}

```

8.実験結果

8-1.分散 σ^2 毎の認識率の比較

以下に,各特徴量を用いた識別器の分散を変えた時のfの値と認識率を示す.

分散	σ^2	100	1000	10000	100000	1000000	10000000
fの値	Piano_1	1.003780	1.014873	1.001436	1.000139	0.999137	0.999587
	Piano_2	0.999372	0.998983	0.999056	0.999990	1.002871	1.005881
	Piano_3	1.061880	1.012575	1.000614	1.000122	1.004168	1.007994
	Piano_4	1.163494	1.028011	1.002367	1.000267	1.003803	1.007363
	Piano_5	0.998676	1.001217	1.000239	0.999937	0.998562	0.998403
	Piano_6	1.000046	1.000659	1.000347	0.999999	1.007986	1.014441
	Piano_7	1.095608	1.011393	1.001416	1.000096	1.006569	1.012027
	Piano_8	1.000573	0.999996	1.000298	0.999961	1.005284	1.009959
	Piano_9	1.084265	1.011198	1.001309	1.000065	1.005558	1.010479
	Piano_10	1.073317	1.011200	1.001209	1.000041	1.005707	1.010816
	Piano_11	1.000048	1.000660	1.000347	0.999999	1.007986	1.014441
	Piano_12	1.147063	1.018840	1.001853	1.000075	1.004589	1.008829
	Piano_13	0.999692	0.998918	0.998994	0.999596	1.001136	1.003314
	Piano_14	1.000045	1.000659	1.000347	0.999999	1.007986	1.014441
	Piano_15	1.000048	1.000660	1.000347	0.999999	1.007986	1.014441
	Piano_16	1.059085	1.006747	1.000935	1.000034	1.007019	1.012737
	Piano_17	1.025841	1.003137	1.000606	1.000018	1.007665	1.013865
	Piano_18	1.024096	1.002827	1.000582	1.000017	1.007720	1.013964
	Piano_19	1.030975	1.003238	1.000628	1.000014	1.007523	1.013636
	Piano_20	1.026770	1.002956	1.000581	1.000005	1.007500	1.013631
	Guitar_1	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000009
	Guitar_2	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_3	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_4	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_5	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_6	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_7	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_8	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_9	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_10	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_11	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_12	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_13	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.386913
	Guitar_14	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_15	-0.999999	-1.000000	-1.000000	-0.999595	-0.999896	-0.999435
	Guitar_16	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-1.000000
	Guitar_17	-0.999999	-1.000000	-1.000000	-0.999595	-1.000004	-2.028628
	Guitar_18	-0.999999	-1.000000	-1.000000	-0.999594	-0.999995	-1.195496
	Guitar_19	-0.999999	-1.000000	-1.000000	-0.999594	-1.000000	-0.999981
	Guitar_20	-0.999999	-1.000000	-1.000000	-0.999594	-0.999931	-0.999485
	認識率[%]	100	100	100	100	100	100

表 1.FFT の特徴量を用いた識別器の認識結果

分散 σ^2	$ f $ の平均値	1との差
100	1.01986635	0.01986635
1000	1.00321868	0.00321868
10000	1.00033778	0.00033778
100000	0.99980638	0.00019363
1000000	1.02766453	0.02766453
10000000	1.04526649	0.04526649

表 2.FFT の特徴量を用いた識別器の $|f|$ の値の平均と 1 との差

表 1 より,分散が 100 から 10000 まではギター音の方が $|f|$ の値が全て同じになっているので,識別器として作用しない可能性がある.なので,分散が 100000 以上の時に最も良い識別器である.次に表 2 より, $|f|$ が最も 1 に近いのは分散が 100000 の時であるので,FFT の特徴量を用いた識別器は分散が 100000 の時に最も認識能力があると考えられる.

分散	σ^2	100	1000	10000
fの値	Piano_1	0.999718	0.999721	0.999994
	Piano_2	0.999751	1.000308	1.007259
	Piano_3	0.999728	0.999910	1.065458
	Piano_4	0.999705	1.000013	1.074681
	Piano_5	0.999648	0.999917	1.142379
	Piano_6	0.999894	1.000242	1.012986
	Piano_7	0.999805	1.000035	1.056060
	Piano_8	0.999673	0.999760	1.119599
	Piano_9	0.999762	0.999938	1.111587
	Piano_10	0.999731	0.999957	1.093859
	Piano_11	1.000022	1.001478	1.050347
	Piano_12	0.999748	1.058286	1.115160
	Piano_13	0.999706	1.000303	1.152003
	Piano_14	0.999883	0.999829	1.001322
	Piano_15	0.999820	0.998958	0.998876
	Piano_16	0.999773	1.000032	1.075692
	Piano_17	0.999692	0.999926	1.101632
	Piano_18	0.999695	1.000008	1.160340
	Piano_19	0.999622	0.999981	1.174129
	Piano_20	0.999620	0.999986	1.173470
	Guitar_1	-0.999883	-1.066455	-1.062307
	Guitar_2	-0.999816	-1.080892	-1.027462
	Guitar_3	-0.999761	-0.999744	-0.999995
	Guitar_4	-0.999838	-1.000097	-1.056547
	Guitar_5	-1.000026	-1.098982	-1.004246
	Guitar_6	-0.999886	-0.999888	-1.007671
	Guitar_7	-0.999768	-1.000089	-1.010486
	Guitar_8	-0.999788	-1.000031	-1.000602
	Guitar_9	-0.999869	-1.000095	-1.000460
	Guitar_10	-0.999791	-1.000317	-1.033569
	Guitar_11	-0.999895	-1.020184	-1.050527
	Guitar_12	-0.999812	-1.048466	-1.068211
	Guitar_13	-0.999755	-1.000049	-1.025696
	Guitar_14	-0.999949	-1.044259	-1.028661
	Guitar_15	-0.999778	-0.999870	-0.999381
	Guitar_16	-0.999872	-0.999782	-0.999379
	Guitar_17	-0.999764	-1.006571	-1.100651
	Guitar_18	-0.999682	-0.999868	-1.060922
	Guitar_19	-0.999691	-1.000156	-1.020080
	Guitar_20	-0.999691	-0.999981	-1.000392
認識率[%]		100	100	100

表 3.ケプストラムの特徴量を用いた識別器の認識結果

分散 σ^2	$ f $ の平均値	1との差
100	0.99968175	0.00031825
1000	1.01060910	0.01060910
10000	1.05610195	0.056102

表 4. ケブストラムの特徴を用いた識別器の $|f|$ の値の平均と1との差

表 3 より, $|f|$ の値が全て同じになることがなく, 表 3 だけでは判断がし難い. 表 4 より, 分散が 100 の時に $|f|$ は最も 1 に近いので, ケブストラムの特徴量を用いた識別器は分散が 100 の時に最も認識能力が高いと考えられる.

分散	σ^2	1000		10000		100000	
		FFT	ケブストラム	FFT	ケブストラム	FFT	ケブストラム
fの値	Piano_1	1.014873	1.000000	0.998235	0.999862	2.755350	3.484386
	Piano_2	0.998983	1.000000	0.999965	1.000095	2.755484	3.508363
	Piano_3	1.012575	1.000000	1.001148	1.037636	2.773468	3.437252
	Piano_4	1.028011	1.000000	1.003276	1.044287	2.787359	3.389809
	Piano_5	1.001217	1.000000	1.003058	1.111954	2.802899	1.823509
	Piano_6	1.000659	0.999999	1.000298	1.012603	2.857312	1.023752
	Piano_7	1.011393	1.000000	1.001796	1.041771	2.850855	1.246828
	Piano_8	0.999996	1.000000	1.000989	1.089130	2.849551	1.757909
	Piano_9	1.011198	1.000000	1.002058	1.087000	2.848593	1.575384
	Piano_10	1.011200	1.000000	1.001908	1.070022	2.849473	1.531344
	Piano_11	1.000660	1.000004	1.000298	1.049901	2.857312	1.063396
	Piano_12	1.018840	1.058275	1.002898	1.097914	2.846358	1.390426
	Piano_13	0.998918	1.000000	1.000320	1.123802	2.833475	1.742751
	Piano_14	1.000659	0.999999	1.000298	1.000732	2.857313	1.001307
	Piano_15	1.000660	0.999998	1.000298	0.999380	2.857311	0.998887
	Piano_16	1.006747	1.000000	1.001127	1.062809	2.855106	1.255268
	Piano_17	1.003137	1.000000	1.000625	1.090140	2.856793	1.263770
	Piano_18	1.002827	1.000000	1.000578	1.147709	2.857003	1.361215
	Piano_19	1.003238	1.000000	1.000643	1.155074	2.856710	1.563354
	Piano_20	1.002956	1.000000	1.000610	1.152160	2.857340	1.609077
	Guitar_1	-1.000000	-1.066452	-1.000000	-1.062373	-1.000000	-1.066802
	Guitar_2	-1.000000	-1.080888	-1.000000	-1.027364	-1.000000	-1.023192
	Guitar_3	-1.000000	-1.000000	-1.000000	-0.999968	-1.000000	-1.000020
	Guitar_4	-1.000000	-1.000000	-1.000000	-1.056428	-1.000000	-1.079830
	Guitar_5	-1.000000	-1.098932	-1.000000	-1.004158	-1.000000	-1.000580
	Guitar_6	-1.000000	-1.000000	-1.000000	-1.007482	-1.000000	-1.037645
	Guitar_7	-1.000000	-1.000000	-1.000000	-1.010518	-1.000000	-1.016272
	Guitar_8	-1.000000	-1.000000	-1.000000	-1.000252	-1.000000	-1.000437
	Guitar_9	-1.000000	-1.000000	-1.000000	-1.000240	-1.000000	-1.016876
	Guitar_10	-1.000000	-1.000000	-1.000000	-1.033811	-1.000000	-1.049839
	Guitar_11	-1.000000	-1.020163	-1.000000	-1.050397	-1.000000	-1.062527
	Guitar_12	-1.000000	-1.048458	-1.000000	-1.068271	-1.000000	-1.074408
	Guitar_13	-1.000000	-1.000000	-1.000000	-1.025667	-1.000000	-1.056837
	Guitar_14	-1.000000	-1.044270	-1.000000	-1.028687	-1.000000	-1.030130
	Guitar_15	-1.000000	-1.000000	-1.000000	-0.999719	-1.000000	-0.999288
	Guitar_16	-1.000000	-1.000000	-1.000000	-0.999669	-1.000000	-0.999386
	Guitar_17	-1.000000	-1.006642	-1.000000	-1.100752	-1.000000	-1.118292
	Guitar_18	-1.000000	-1.000000	-1.000000	-1.061293	-1.000000	-1.107291
	Guitar_19	-1.000000	-1.000000	-1.000000	-1.020287	-1.000000	-1.031648
	Guitar_20	-1.000000	-1.000000	-1.000000	-1.000221	-1.000000	-1.000461
	認識率[%]	100		100		100	

表 5. FFTとケブストラム両方の特徴量を用いた識別器の認識結果

分散 σ^2	$ f $ の平均値	1との差
1000	1.00691034	0.00691034
10000	1.02439955	0.02439955
100000	1.66693516	0.66693516

表 6.FFTとケプストラム両方の特徴量を用いた識別器の $|f|$ の値の平均と1との差

表 5 より,ギター音の FFT による特徴量を用いた時は $|f|$ はどの分散でも同じ値となった.表 6 より, $|f|$ が最も 1 に近いのは分散が 1000 の時であったので,FFT とケプストラム両方の特徴量を用いた識別器は分散が 1000 の時に最も認識能力が高いと考えられる.

8-2.分散 σ 二乗毎の正解率の比較

以下に,各特徴量を用いた識別器の分散を変えた時の f の値と正解率を示す.

分散	σ^2	100	1000	10000	100000	1000000	10000000
fの値	piano_test_1	1.037974	1.006788	1.000741	0.999986	1.007698	1.014047
	piano_test_2	1.008645	1.001891	1.000440	1.000000	1.007930	1.014347
	piano_test_3	0.059599	0.756039	0.972400	0.997070	1.006829	1.012850
	piano_test_4	0.825260	0.982034	0.998426	0.999791	1.007800	1.014184
	piano_test_5	0.995634	1.000549	1.000308	0.999988	1.007939	1.014376
	piano_test_6	1.011886	1.002372	1.000475	1.000001	1.007919	1.014337
	piano_test_7	0.998096	1.000459	1.000328	0.999997	1.007959	1.014393
	piano_test_8	1.024290	1.003458	1.000616	1.000021	1.007712	1.013983
	piano_test_9	1.030994	1.005601	1.000795	1.000042	1.007776	1.014086
	piano_test_1	1.074892	1.012116	1.001420	1.000113	1.007144	1.013022
	guitar_test_1	0.000000	0.000000	0.000000	-0.000001	-0.248524	-1.803127
	guitar_test_2	0.000000	0.000000	0.000000	-0.000235	-0.388217	-1.021470
	guitar_test_3	0.000000	0.000000	0.000000	-0.000003	-0.209985	-0.560193
	guitar_test_4	0.000000	0.000000	0.000000	0.015097	0.389166	0.490277
	guitar_test_5	0.000000	0.000000	0.036447	0.717283	0.927729	0.930687
	guitar_test_6	0.000000	0.000000	0.000000	0.000000	-0.001787	0.034534
	guitar_test_7	0.000000	0.000000	0.000000	0.000000	0.147600	0.616510
	guitar_test_8	0.000000	0.000000	0.000000	0.084005	0.711660	0.895626
guitar_test_9	0.000000	0.000000	0.000000	0.011466	0.530227	0.857863	
guitar_test_1	0.000000	0.000000	0.000000	0.129024	0.733360	0.906205	
	正解率[%]	50	50	50	65	70	65

表 7.FFT の特徴を用いた識別器の識別結果

表 7 より,ピアノの場合はよく識別されているが,ギターの識別があまりされてことがわかる.分散 1000000 の時に正解率が最も高く,ピアノの特徴量を入力した場合の $|f|$ も 1 を越えているので,最も識別能力があると考えられる.

分散	σ^2	100	1000	10000
fの値	piano_test_1	0.008568	0.782620	1.122581
	piano_test_2	0.000753	0.650500	1.107808
	piano_test_3	0.007441	0.834008	1.233909
	piano_test_4	0.002705	0.670675	1.235688
	piano_test_5	0.000042	0.601544	1.141405
	piano_test_6	0.000341	0.628584	1.027224
	piano_test_7	0.000033	0.597917	1.141340
	piano_test_8	0.062114	0.900928	1.069556
	piano_test_9	0.068587	0.984670	1.094049
	piano_test_1	0.064189	1.011621	1.076863
	guitar_test_1	-0.055137	-0.975438	-1.009839
	guitar_test_2	-0.025772	-0.846247	-1.074294
	guitar_test_3	-0.001601	-0.547482	-0.860604
	guitar_test_4	-0.008565	-0.678491	-0.901342
	guitar_test_5	-0.015671	-0.752556	-0.925068
	guitar_test_6	-0.027055	-0.933977	-1.022000
	guitar_test_7	-0.014516	-0.856628	-1.021346
	guitar_test_8	-0.036083	-0.859628	-0.981045
guitar_test_9	-0.026735	-0.914442	-0.961585	
guitar_test_1	-0.019283	-0.816580	-0.973881	
	正解率[%]	100	100	100

表 8. ケプストラムの特徴量を用いた識別器の識別結果

表 8 より, ケプストラムの場合は全ての分散の値によって正解率が 100%であった. その中でも, 分散が 10000 の時に最も多く f の値が 1 を越えているので, 最も識別能力が高いと考えられる.

分散	σ^2	1000		10000		100000	
		FFT	ケプストラム	FFT	ケプストラム	FFT	ケプストラム
fの値(FFT)	piano_test_1	1.006788	0.782628	1.000505	1.109616	2.855398	1.340909
	piano_test_2	1.001891	0.650432	1.000361	1.100095	2.856475	1.223077
	piano_test_3	0.756039	0.834014	0.971988	1.217259	2.821148	1.582461
	piano_test_4	0.982034	0.670605	0.998352	1.225245	2.851039	1.450199
	piano_test_5	1.000549	0.601556	1.000289	1.135541	2.856714	1.245287
	piano_test_6	1.002372	0.628436	1.000385	1.020650	2.856662	1.114829
	piano_test_7	1.000459	0.597737	1.000285	1.135583	2.857599	1.247907
	piano_test_8	1.003458	0.900897	1.000340	1.058473	2.854972	1.214005
	piano_test_9	1.005601	0.984672	1.000783	1.079320	2.853870	1.315323
	piano_test_1	1.012116	1.011619	1.000964	1.061263	2.844056	1.306362
	guitar_test_1	0.000000	-0.975444	0.000000	-1.009876	-0.000001	-1.011849
	guitar_test_2	0.000000	-0.846236	0.000000	-1.074473	-0.000235	-1.119455
	guitar_test_3	0.000000	-0.547572	0.000000	-0.860892	-0.000003	-0.923136
	guitar_test_4	0.000000	-0.678576	0.000000	-0.901473	0.039177	-0.935670
	guitar_test_5	0.000000	-0.752611	0.036070	-0.925172	1.925195	-0.954099
	guitar_test_6	0.000000	-0.934003	0.000000	-1.021900	0.000000	-1.029521
	guitar_test_7	0.000000	-0.856657	0.000000	-1.021412	0.000001	-1.037810
	guitar_test_8	0.000000	-0.859660	0.000000	-0.980907	0.245917	-0.998211
guitar_test_9	0.000000	-0.914423	0.000000	-0.961305	0.031349	-0.961503	
guitar_test_1	0.000000	-0.816579	0.000000	-0.973648	0.324063	-0.996209	
	正解率[%]	75		75		82.5	

表 9. FFT とケプストラム両方の特徴量を用いた識別器の識別結果

表 9 より,分散が 1000 と 10000 の時はギター音の FFT による特徴量を入力しても識別がされていないので,分散が 100000 の時に最も識別能力があると考えられる.

8-3.各識別器の認識率と正解率の比較

8-1,8-2 より,分散 σ^2 の違いにより各識別器の中で最も良い識別器であると考えられるものを求めたが,認識率を重視した場合と正解率を重視した場合で分散の値が異なっていたため,现阶段では最も良い超平面がどれか決定することができなかった.なので,ここでは正解率を重視し,正解率が最も高い識別器を各特徴量毎に比較する.

特徴量	認識率[%]	正解率[%]
FFT	100	70
ケプストラム	100	100
FFT+	100	82.5
ケプストラム		

表 10.それぞれの特徴量を用いた識別器の認識率と正解率の比較

認識率はどの特徴量を用いても 100%であったことから,分離超平面はどの特徴量を用いても求められることがわかる.ただし,正解率はケプストラムは 100%であったが,FFT が低いいため 2 種類の特徴量を組み合わせた識別器は正解率が下がった.これは,FFT の場合高周波成分がほぼ 0 となるので,この部分の特徴量を使用してもあまり効果が見られないためだと考えられる.

9.まとめ

ピアノとギターの識別器を作成することができた。パワースペクトルよりもケプストラムの特徴量を用いた方が識別器には有用であることがわかった。FFTの正解率があまり良くなかった原因として、高周波成分がほぼ0となり、音高情報にあたる特徴量が使用できないため、学習データとして機能しなかったからだと考えられる。それに対しケプストラムは、楽器情報と音高情報が分離でき、両方の特徴量を使用できたため、高い正解率を示したと考えられる。しかし、今回は鍵盤楽器であるピアノと弦楽器であるギターの識別であったため、波形の違いも大きく、ケプストラムでも上手く識別できたのではないかと思う。鍵盤楽器同士、弦楽器同士では上手く識別できるのか検討したい。しかし、全体的に見ると、認識率、正解率共に高い割合を示したため、良い結果が得られた。

今後の課題として、認識率と正解率で最も適した分散が異なり、真の分離超平面を求めることはできなかったため、分散が異なった原因を解明し、分離超平面を求める。そして、学習データとテストデータの個数が少なかったので個数を増やして実験を行い、また、MFCCを作成し、MFCCによる結果とケプストラムによる結果との比較を行う。

10.参考文献

- [1]北原聡志,甲藤二郎:Supprot Vector Machine を用いた楽器同定における特徴量の比較,電子情報通信学会総合大会,2005
- [2]Curtis Roads 著,青柳龍也 訳:コンピュータ音楽(東京電機大学出版局,2001)
- [3]酒井幸市 著:デジタル信号処理(コロナ社,1996)
- [4]坂巻佳壽美 著:見てわかるデジタル信号処理(工業調査会,1998)
- [5]鹿野清宏 著:音声認識システム(オーム社)
- [6]今井聖 著:音声信号処理(森北出版株式会社,2005)
- [7]音声認識のしくみ
<http://recognition.web.fc2.com/>
- [8]Nello Cristianini 著,大北剛 訳:サポートベクター入門(共立出版株式会社,2005)
- [9]渡辺澄夫 著:学習システムの理論と実現(森北出版株式会社,2005)
- [10]甘利俊一 著:パターン認識と学習の統計学(岩波書店,2003)
- [11]渋谷道雄 著:マンガでわかるフーリエ解析(オーム社,2006)