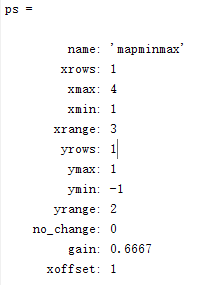
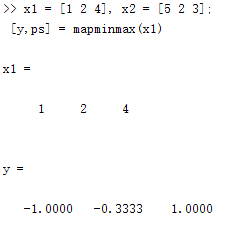
**这周主要读了两篇的论文，其中第一篇是提取音乐的特征向量并用神经网络进行音乐分类，这一篇着重了解关于数据的归一化方法，论文中也有提到相应的代码，但是由于太复杂还没仔细研究。第二是看到了老师给的关于mfcc,就去网上查找了相关的资料，了解梅尔倒谱系数的是什么，提取的过程，以及提取后的用处，相关的代码实现也做了一下仿真，理解的地方相应的做了标注，但是对于整体的代码还未做到融汇贯通，真正把握，因此下周想接着往这个方向查阅研究下去，先了解特征是怎样提取，再进一步了解怎么进行分类。**

1. **《音乐分类及其 Matlab 实现》**
2. 目的

利用不同类别音乐的统计规律提取特征向量，并利用神经网络进行分类。

1. 归一化

通过论文，了解了归一化这个内容，归一化就是要把你需要处理的数据经过处理后（通过某种算法）限制在你需要的一定范围内，由于采集的各数据单位不一致，因而须对数据进行[-1，1]归一化处理，例如以下小程序，mapminmax将矩阵每一行进行归一化。



y是对进行某种规范化后得到的数据，这种规范化的映射记录在结构体ps中，也就是说对x1 = [1 2 4]采用这个映射 f: 2\*(x-xmin)/(xmax-xmin)+(-1),就可以得到y = [ -1.0000 -0.3333 1.0000]，在神经网络和许多的分类中都会使用到归一化这个方法，使数据变得简单，在matlab中，归一化有以下几种方法：

pre\*\*是归一化，post\*\*是反归一化，tram\*\*是使用同样的设置归一化另外一组数据。

1. premnmx、tramnmx、postmnmx

**premnmx函数，**用于将网络的输入数据或输出数据进行归一化，归一化后的数据将分布在[-1,1]区间内，[Pn,minp,maxp,Tn,mint,maxt]=premnmx(P,T)，其中P，T分别为原始输入和输出数据，minp和maxp分别为P中的最小值和最大值，mint和maxt分别为T的最小值和最大值。

**tramnmx函数**，在训练网络时如果所用的是经过归一化的样本数据，那么以后使用网络时所用的新数据也应该和样本数据接受相同的预处理，tramnmx语句的语法格式是：[PN]=tramnmx(P,minp,maxp)，其中P和PN分别为变换前、后的输入数据，maxp和minp分别为premnmx函数找到的最大值和最小值。

**postmnmx函数，**将网络输出结果进行反归一化还原成原始的数据，postmnmx语句的语法格式是：[PN] = postmnmx(P,minp,maxp)，其中P和PN分别为变换前、后的输入数据，maxp和minp分别为premnmx函数找到的最大值和最小值。

1. prestd、poststd、trastd

归化数据到(0,1) ，与（1）差不多，每个函数的方法可以用matlab help\*\*来查阅。

1. 是用matlab语言自己编程

但我发现挺多的代码归一化都是使用自己的编程方法来实现

1. **语音特征参数MFCC提取过程详解**

**文章参考网址**

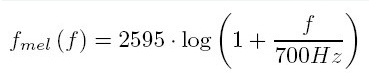
<http://blog.csdn.net/ziyuzhao123/article/details/20629323>

<http://www.mamicode.com/info-detail-478145.html>

<http://blog.sciencenet.cn/blog-907554-967295.html>

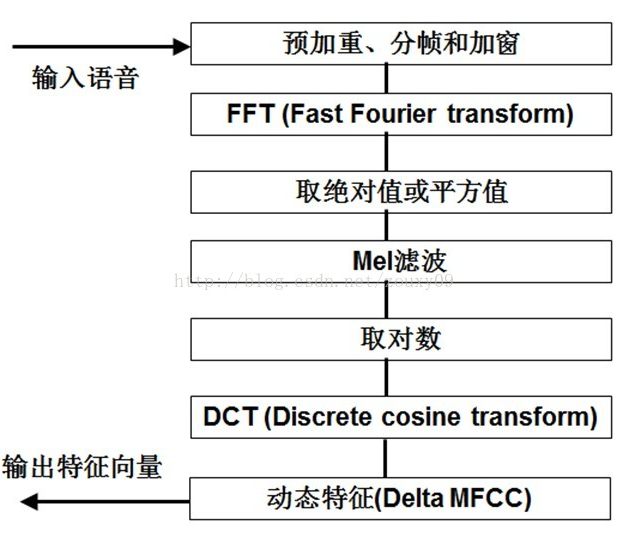
首先先了解了一下什么是mfcc梅尔倒谱系数。在语音识别（Speech Recognition）和话者识别（Speaker Recognition）方面，最常用到的语音特征就是梅尔倒谱系数。Mel频率是基于人耳听觉特性提出来的。人通过声道产生声音，声道的形状决定了发出怎样的声音。声道的形状在语音短时功率谱的包络中显示出来，而MFCCs就是一种准确描述这个包络的一种特征。

**倒谱（cepstrum）**就是一种信号的傅里叶变换经对数运算后再进行傅里叶反变换得到的谱。**Mel频率**描述了人耳频率的非线性特性，它与频率的关系可用下式近似表示，其中f为频率，单位Hz



**梅尔倒谱系数是**Mel频谱上面获得的倒谱系数

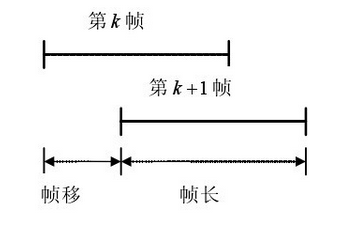
提取流程（在网址中可以看到详细的介绍）



**预加重、分帧和加窗；（加强语音信号性能（信噪比，处理精度等）的一些预处理）**

1. **预加重：**预加重的目的是提升高频部分，使信号的频谱变得平坦，能用同样的信噪比求频谱。
2. **分帧：**之后要对语音信号进行傅里叶变换，但是音频信号是时变的，因此要分成一小段一小段进行积分。enframe（）就是先将N个采样点集合成一个观测单位，称为帧。通常情况下N的值为256或512，涵盖的每帧长度约为10~30ms左右。以8KHz来说，若帧长度为256个采样点，则对应的时间长度是256/8000×1000=32ms。

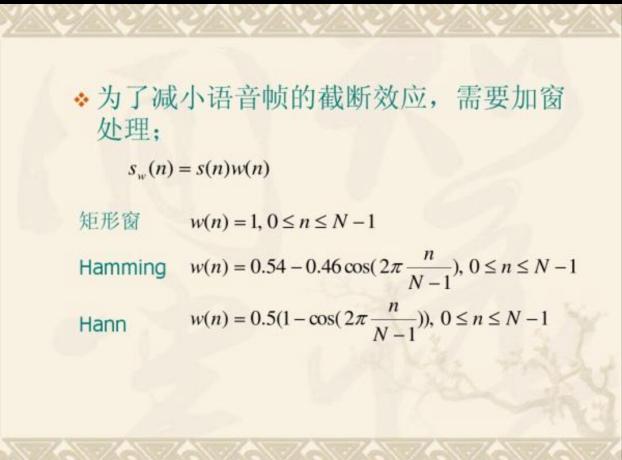
帧移指前后两帧的重叠量，即前一帧尾部与后一帧头部的重叠量，防止两帧之间的不连续。

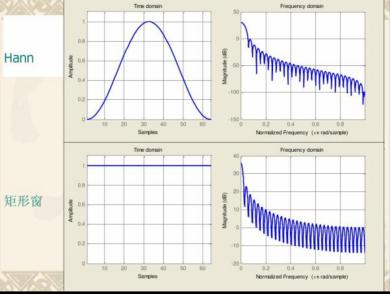
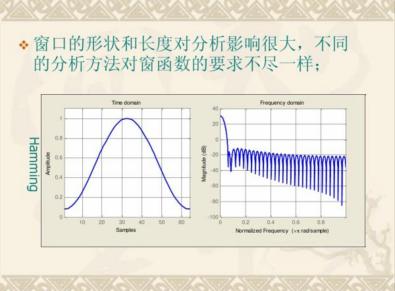
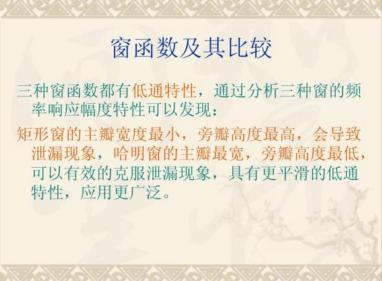


1. **加窗：**将每一帧乘以汉明窗，减小语音帧的截断效应，以增加帧左端和右端的连续性。假设分帧后的信号为S(n), n=0,1…,N-1, N为帧的大小，不同的a值会产生不同的汉明窗，一般情况下a取0.46，那么乘上汉明窗后

IMG_256

IMG_256





1. **快速傅里叶变换：**对分帧加窗后的各帧信号进行快速傅里叶变换得到各帧的频谱。并对语音信号的频谱取模平方得到语音信号的功率谱。x(n)为输入的语音信号，N表示傅里叶变换的点数。

IMG_256

1. **三角带通滤波器：**通过Mel频谱，将线形的自然频谱转换为体现人类听觉特性的Mel频谱。

6.**在Mel频谱上面进行倒谱分析（取对数，做逆变换，实际逆变换一般是通过DCT离散余弦变换来实现），取DCT后的第2个到第13个系数作为MFCC系数，获得Mel频率倒谱系数MFCC，这个MFCC就是这帧语音的特征。**

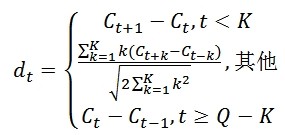
计算每个滤波器组输出的对数能量：

IMG_256

离散余弦变换（DCT）

IMG_256

1. **动态差分参数的提取（包括一阶差分和二阶差分）：**标准的倒谱参数MFCC只反映了语音参数的静态特性，语音的动态特性可以用这些静态特征的差分谱来描述，差分参数的计算可以采用下面的公式，dt表示第t个一阶差分，Ct表示第t个倒谱系数，Q表示倒谱系数的阶数，K表示一阶导数的时间差，可取1或2。将上式的结果再代入就可以得到二阶差分的参数。



这时候，语音就可以通过一系列的倒谱向量来描述了，每个向量就是每帧的MFCC特征向量，这样就可以通过这些倒谱向量对语音分类器进行训练和识别了。

1. **代码实现**

这段代码可能并不是读的很透，有的理解部分做了自己的解释

%读取音乐文件，可将音乐直接放到与代码同一目录下，则可不用写路径

[x,fs]=audioread('sun1.wav');

%播放音乐

sound(x, fs);

%画出原始的波形

figure(1)

plot(x),title('音乐的时域波形')

%构造Mel滤波器，滤波器阶数为24，FFT变换的长度为256，采样频率fs为16000Hz ，0到0.4？

bank=melbankm(24,256,fs,0,0.4,'t'); %得到一个矩阵

%归一化Mel滤波器组系数 ，归一化的方法是用matlab自己编程

bank=full(bank); %full()将稀疏矩阵转换为全矩阵

bank=bank/max(bank(:)); %bank(:)表示把bank矩阵按顺序变成一列

for k=1:12

n=0:23;

dctcoef(k,:)=cos((2\*n+1)\*k\*pi/(2\*24)); %？

end

w=1+6\*sin(pi\*[1:12]./12);%归一化倒谱提升窗口

w=w/max(w);%预加重滤波器

xx=double(x);%为什么变成双精度

xx=filter([1-0.9375],1,xx);%语音信号分帧

xx=enframe(xx,256,80);%对xx 256点分为一帧

%计算每帧的MFCC参数

for i=1:size(xx,1)

y=xx(i,:);

s=y'.\*hamming(256); %加窗处理，每一个帧采样点 的语音信号乘以汉明窗

t=abs(fft(s));%FFT快速傅里叶变换

t=t.^2;

c1=dctcoef\*log(bank\*t(1:129));

c2=c1.\*w';

m(i,:)=c2;

end

%以下部分为提取动态特征，提高系统的识别性能%

%求一阶差分系数

dtm=zeros(size(m));

for i=3:size(m,1)-2

dtm(i,:)=-2\*m(i-2,:)-m(i-1,:)+m(i+1,:)+2\*m(i+2,:);

end

dtm=dtm/3;

%求取二阶差分系数

dtmm=zeros(size(dtm));

for i=3:size(dtm,1)-2

dtmm(i,:)=-2\*dtm(i-2,:)-dtm(i-1,:)+dtm(i+1,:)+2\*dtm(i+2,:);

end

dtmm=dtmm/3;

%合并mfcc参数和一阶差分mfcc参数

ccc=[m dtm dtmm];

%去除首尾两帧，以使两帧的一阶差分参数为0

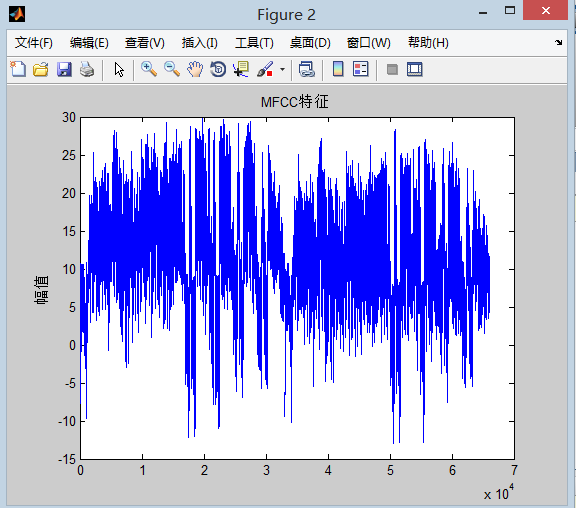
ccc=ccc(3:size(m,1)-2,:);

ccc;

figure(2)

ccc\_1=ccc(:,1);

plot(ccc\_1);title('MFCC特征');ylabel('幅值');



（1）归一化mel滤波器组系数时需要的melbankm函数在matlab中是不存在的，因此通过网上的查询说需要下一个voicebox添加到matlab中，voicebox用于语音编码语音压缩和语音识别，下载方式如下

官方下载：

<http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.zip>

1、解压voicebox.zip，将整个目录voicebox复制到MATLAB的安装目录的TOOLBOX目录下，如D:\MATLAB7\toolbox\

2、（方法一）打开Matlab，在MATLAB命令窗口中输入以下命令：

 >>cd D:\MATLAB7\toolbox\voicebox

 >>addpath(genpathKPM(pwd))

（方法二）将TOOLBOX下新加的voicebox工具箱加到MATLAB的搜索路径中去。添加voicebox工具箱的MATLAB的搜索路径也可采用如下指令

>>addpath(genpath('D:\MATLAB7\toolbox\voicebox'))

1. 为了永久保存上面的路径，以免下次重启MATLAB时重新添加，在MATLAB命令窗口下使用下面的命令：

>>savepath

4、检验是否成功设置的方法：在命令窗口中输入以下命令：

Which melbankm.m（可以为所加工具箱的任一个M文件名称），如果显示正确，就说明上面的设置成功。

 >>which melbankm.m

  D:\MATLAB7\toolbox\voicebox\melbankm.m