HCS 7367
Speech Perception Lab

Dr. Peter Assmann

Fall 2011

Sine wave synthesis

- Generate a 500-Hz pure tone

% 100 ms duration; 60 dB; 10 kHz sample rate)

```
>> dur=100; % duration in ms
>> db=60; % amplitude in dB re: 1
>> amp=10.^(db/20); % convert amplitude from dB to linear
>> freq=500; % frequency of tone in Hz
>> Fs=10000; % sample rate in Hz
>> T=1/Fs; % sampling interval
>> n=dur*Fs/1000; % duration in samples
>> phase = 0; % phase
>> y=amp .* cos(2*pi*T*(1:n)*freq+phase); % vector y = synthesized tone
```

Sine wave synthesis

- Generate 5-component complex with equal amplitudes and frequencies of 200, 400, 600, 800 and 1000 Hz; 100 ms duration; 60 dB; 10 kHz sample rate)

```
% All variables as above except:
db=[60 60 60 60 60];
amp=10.^(db/20);
freq=[200 400 600 800 1000];
phase = [0 0 0 0 0];
```

% introduce a **for-loop**:

```
y=zeros(1,n);
nharm=length(freq);
for i=1:nharm
   harmi=amp(i).*cos(2*pi*T*(1:n)*freq(i)+phase(i));
   y=y+harmi;
end;
```

Group project

- Modify the script to demonstrate that a square wave can be constructed by summing odd harmonics with amplitudes 1/freq.

```
dur=100;
amp=[ … ];
freq=[ … ];
phase = [ …];
Fs=10000;
T=1/Fs;
n=dur*Fs/1000;
```

```
y=zeros(1,n);
nharm=length(freq);
for i=1:nharm
   harmi=amp(i).*cos(2*pi*T*(1:n)*freq(i)+phase(i));
   y=y+harmi;
end;
```

Group project

- Modify the script to synthesize a steady-state approximation to the vowel /i/. Estimate the amplitudes from the amplitude spectrum of a real vowel and set the frequencies to [100, 200, … 100*nharm].

```
dur=100;
amp=[ … ];
freq=[ … ];
phase = [ …];
Fs=10000;
T=1/Fs;
n=dur*Fs/1000;
```

```
y=zeros(1,n);
nharm=length(freq);
for i=1:nharm
   harmi=amp(i).*cos(2*pi*T*(1:n)*freq(i)+phase(i));
   y=y+harmi;
end;
```

Group project

1. Measure the center frequencies of the formant frequencies F1, F2 and F3 from a 20-ms time segment sampled at the midpoint of each of your 12 vowels.
2. Check your measurements by comparing any two of these methods: Wavesurfer, Praat and TrackDraw.

Group project

3. Store formant measurements (as integer values, rounded to nearest whole number) in an Excel file (3 columns x 12 rows).

F1	F2	F3
567	1503	2976
435	1456	3041
...	...	...

4. Load the Excel file into Matlab.
   - [num,txt,raw] = xlsread('F1F2F3_data.xlsx')

Group project

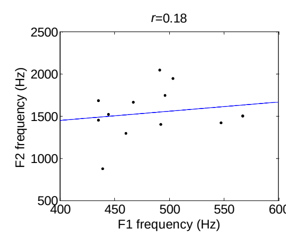
5. Load the Excel file into Matlab.
   - [num,txt,raw] = xlsread('F1F2F3_data.xlsx');
   - F1 = num(:,1);
   - F2 = num (:,2);
   - F3 = num (:,3);
6. Make a scatterplot of F1 x F2.
   - plot(F1,F2, 'o');

Group project

7. Carry out a linear regression predicting F2 scores from F1.
   - [slope,yint,pcorr]=lsq(F1,F2,1);
   - title('\itr\rm=0.18'); % show Pearson's *r* in title
   - xlabel('F1 frequency (Hz)')
   - ylabel('F2 frequency (Hz)')

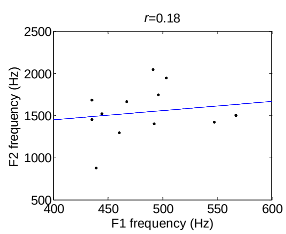
Group project

8. Superimpose regression line on scatterplot:
   - x= [400 600];
   - yhat=slope*x + yint;
   - hold on;
   - plot(x,yhat);



Group project

9. Automatically calculate x from x-axis range:
   - x=min(get(gca,'XLim')) max(get(gca,'XLim'))
   - yhat=slope*x + yint;
   - hold on;
   - plot(x,yhat);



Statistical pattern recognition

- Discriminant function analysis (DFA)
  - Technique for classification of objects based on measurements
    - diagnosis of medical conditions
    - assignment of individuals to taxonomic categories (e.g., animal or plant varieties);
    - classification of speech sounds based on acoustic measurements
    - pattern recognition model - neural network implications

Discriminant function analysis

- Given a set of measurements of the acoustic properties of recorded vowel tokens, DFA assigns each token to the vowel category it most resembles (making it possible to identify tokens that would be misclassified).

Discriminant function analysis

- DFA incorporates a category assignment rule: *assign an incoming token to the vowel category which it most closely resembles* (e.g., a vowel with formant frequencies of 300, 2500, 3200 Hz corresponds to the vowel /i/)

Discriminant function analysis

- DFA also provides the estimated *a posteriori* probability of group membership, $p(g_i|x)$ (where g_i is the i^{th} vowel category, and x is the vector of measurements for the token).

Discriminant function analysis

- The *a posteriori* probability of x belonging to group i is:

$$p(i \mid x) = \frac{\exp(-d_i^2(x))}{\sum_{n=1}^{p} \exp(-d_n^2(x))}$$

Discriminant function analysis

- DFA constructs a *discriminant criterion,* based on measurements in the training set, to classify observations in a second set of measurements, x, that define the test set.

Discriminant function analysis

- The *discriminant criterion* is based on a measure of *generalized squared distance* d_i^2 from x to group i is computed as:

$$d_i^2(x) = (x - mi)' S_i - 1 (x - mi)$$

where x is the vector of measurements of dimension p; mi is a vector of dimension p, containing the means of the measurements for group i; S_i is the pooled covariance matrix for group i.

Classification of speech sounds

1. Phoneme classification
2. Recognition of words in connected speech
3. Talker identification (gender, age, dialect)
4. Language recognition
5. Emotional state

Statistical Pattern Recognition

- Techniques for grouping the members of a set of unlabeled items (called the *test set*) into two or more *classes* by reference to a set of labeled items (called the *training set*) whose category membership is known.

Parameters or Features

- **Parameters** or **features** are acoustic properties that can be measured and used to compare a member of the test set with members of the training set for the purpose of category assignment.
- Normally speech sounds differ along several dimensions and sets of features (*feature vectors* or *parameter vectors*) are required for classification.

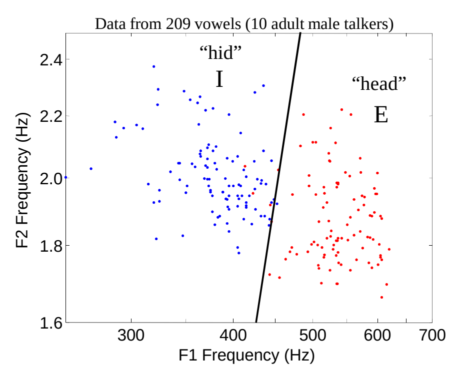
Discriminant function analysis (DFA)

- Given a set of measurements of the acoustic properties of recorded vowel tokens, DFA assigns each token to the vowel category it is most likely to belong to (making it possible to identify tokens that would be misclassified).

Linear DFA

- **Step 1**: *Generate training data*
  - Make a set of F1 and F2 measurements of vowels to use as training data
  - [Plot the measurements in a 2-dimensional plane, or "vowel space"]
  - Find a set of *linear boundaries* in the training data that optimally divide up the vowel space into separate regions for each vowel

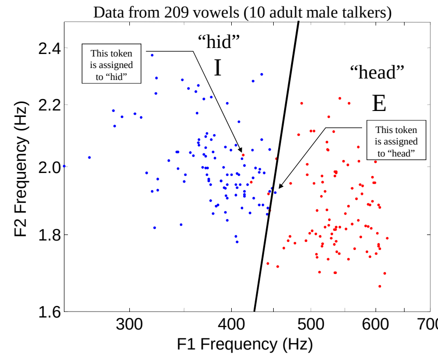
Vowel classification: 2 categories



Linear DFA

- **Step 2**: *Category assignment for test data*
  - Make F1 and F2 measurements of test data
  - [Plot test measurements in the “vowel space”]
  - Assign category labels based on the *linear boundaries* assigned in the training stage

Vowel classification: 2 categories



Vowel space

- An incoming vowel token is projected in the *N*-dimensional vowel space defined by the measurements.
- When N>3, visual separation is difficult.

Distance measures

- Because the same vowel spoken by different talkers or in different contexts vary in their acoustic properties, we need to develop a measure of distance between points in the vowel space.

Distance measures

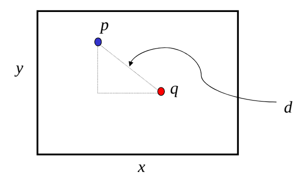
- **Euclidean distance**
  - The Euclidean distance *d* between points *p* and *q* in a 2-dimensional space is given by

$$d = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}$$

Distance measures

- **Example**

$p_x=250$
$q_x=500$
$p_y=1500$
$q_y=1200$



$$d = \sqrt{(500-250)^2 + (1500-1200)^2} = 390.5$$

Distance measures

- **Euclidean distance**
  - The distance d between the **parameter vectors** $\boldsymbol{p}$ and $\boldsymbol{q}$ (each with N elements) is given by

$$d = \sqrt{\sum_{n=1}^{N} (p_n - q_n)^2}$$

Exercise

- Given this table of formant frequencies for the vowels /i/, /a/, and /u/, use Matlab to calculate the Euclidean distances between all pairs of the three vowels.

	F1	F2	F3
/ i /	250	2250	3050
/a/	650	1050	2950
/u/	350	1150	1950

Exercise

- Include the Matlab code with your table.
- Hint: because distances are symmetrical, your output table should look like this:

	/ i /	/a/	/u/
/ i /		*x*	*y*
/a/			*z*
/u/			

Category assignment

- Assign each token in the test set to the category in the training set it is most likely to represent.
  - Compute the parameter vector for the unknown token.
  - Compare it to the parameter vectors for each category (type) in the training set.
  - Compute the *distance* (similarity) between the unknown token and all members of the category in the training set.
  - Assign the token to the category that yields the smallest distance.

Samples and populations

- How do we compare the parameter vector of the unknown token to the parameter vectors of a set of known tokens representing n categories?
- Using probability theory we can describe the measurements as *samples* from a *population*.

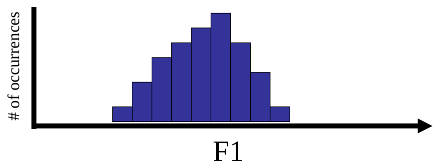
Probability distributions

- In the absence of other knowledge, we can estimate the *a priori* probability that an unknown token belongs to the i^{th} group:

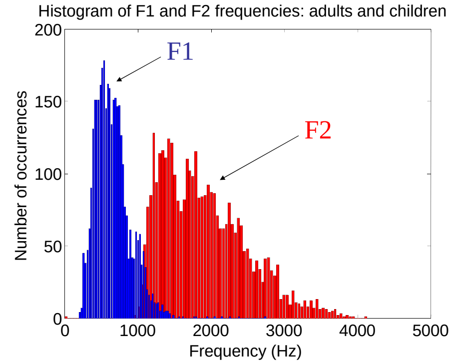
$$p(g_i) = \frac{Number\ of\ tokens\ of\ g_i}{Total\ number\ of\ tokens}$$

Probability density

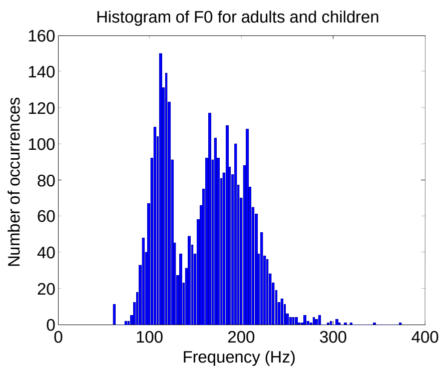
- For a given parameter vector (e.g. F1) we can estimate the probability of observing a given value of that variable in the population.



Probability density



Probability density



Class conditional probability

- The probability of observing a given parameter value p_j given that the token is known to belong to category g_i is called the class conditional probability: $p(\, p_j \,|\, g_i \,)$.

Class conditional probability

- The measure $p(\, p_j \,|\, g_i \,)$ is read as "the probability of p_j given g_i" … for example, the probability that F1 has a given frequency for tokens belonging to the vowel category /i/.
- $p(\, p_j \,|\, g_i \,)$ is often represented as a probability density (showing the conditional probability as a function of the parameter value).

Class conditional probability

- The prior probability for a parameter value $p(\, p_j \,)$ can be calculated by summing the conditional probabilities over all categories and multiplying by the prior probability for the category:

$$p(p_j) = p(p_j \,|\, g_i)\, p(g_i)$$

Posterior probability

- The posterior probability that a token belongs to group g_i (given a value of parameter p_j) is written $p(g_i \mid p_j)$
  - In this case we know the value of the parameter p_j for the token in question, but do not know what category it belongs to.

Bayes' theorem

- The quantity $p(g_i \mid p_j)$ can be estimated from sample data plus knowledge of prior probabilities using Bayes' theorem:

$$p(g_i \mid p_j) = \frac{p(p_j \mid g_i)\, p(g_i)}{p(p_j)}$$

Gaussian distribution

- Accurate estimates of conditional probabilities require large sample sizes. Histograms based on small samples provide poor estimates of the true probability distribution. However improved predictions can be made if we can assume a normal or Gaussian distribution of the probabilities. This assumption requires that the data come from a random sample drawn from a randomly distributed variable.

Gaussian distribution

- A normal distribution is characterized by two quantities, its mean (average or expected value) and standard deviation (dispersion around the mean).
- The quantity σ^2 is the variance:

$$\sigma^2 = \frac{\sum_{i=1}^{N} (v_i - \mu)^2}{N}$$

Gaussian distribution

- Given a Gaussian probability density with mean μ and variance σ^2, the probability of a parameter value v for a token belonging to category g is given by

$$p(v \mid g) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-\frac{(v-\mu)^2}{2\sigma^2}}$$

Gaussian distribution

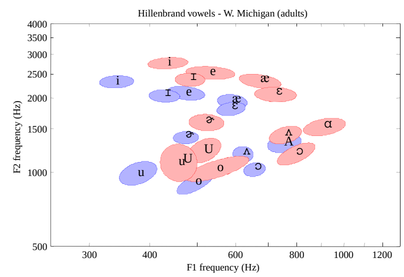
- **Exercise**: Use Matlab to calculate the conditional probability of observing an F1 value v=360 Hz for the vowel g=/I/. The estimated population mean is μ=387 Hz and the estimated population standard deviation is σ=79 Hz.

  p(360 | /I/) = _________

Gaussian distribution

- In the exercise, there was only one parameter (F1) but we know that speech categories are generally distinguished by several features, hence we need more than one parameter vector.
- When more than one parameter vector is used, the mean becomes a vector of values (called the centroid or center of gravity) and the variance becomes a covariance matrix.

Vowel ellipses



Pattern recognition studies

- Statistical pattern recognition techniques
  - Linear vs quadratic DFA
  - Mahalanobis distance
  - Jacknife techniques
  - Data reduction methods
  - PCA

Discriminant function analysis

- DFA constructs a *discriminant criterion*, based on measurements in the training set, to classify observations in a second set of measurements, x, that define the test set.

Discriminant function analysis

- The *discriminant criterion* is based on a measure of *generalized squared distance* d_i^2 from v to group i is computed as:

$$d_i^2(v) = (v - \mu_i)' S_i^{-1} (v - \mu_i)$$

where v = vector of measurements;

μ_i = vector of means for group i;

S_i is the pooled covariance matrix for group i.

Exercise

- Compute Euclidean distances between your measured vowel samples and the formant means for N. Central Texas:

http:www.utdallas.edu/~assmann/hcs7367/formant_data.html

- Assign each token to the category for which distance is smallest.
- Analyze the pattern of errors.