Problem Set 4

**important:** Problems 1 through 3 will be checked for accuracy with a driver program. Failure to follow the interface will be penalized heavily. Please see the note after problems 1 through 3 for help on debugging these problems.

1. (20 points) Write a function to compute the probability of a multivariate Gaussian. It should have the following signature:

   \[ P = f_{gauss}(\text{ColVectors}, \mu, \Sigma) \]

   Returns the likelihood for each vector in a set of column vectors given a normal distribution (Gaussian) with mean \( \mu \) (column vector) and variance-covariance matrix \( \Sigma \). \( P \) is a column vector where \( P(i) = f_{gauss}(\text{ColVectors}(:, i), \mu, \Sigma) \).

   It is common to restrict variance-covariance matrices to be diagonal as this greatly simplifies the computation and we usually try to find features that are independent from one another. Include optimizations for diagonal covariance matrices (the determinant of \( \Sigma \) is simply the product of the diagonal elements, \( \Sigma^{-1} \) is \( 1/\Sigma \), and the matrix multiplication is greatly simplified.

2. (20 points) Write function \( gmmInit \) that initializes a GMM from a set of column oriented training data with the following signature:

   \[ \text{gmm} = gmmInit(\text{Mixtures}, \text{TrainingData}, \text{DiagonalCovar}, \text{Codebook}) \]

   Initialize a Gaussian mixture model with the specified number of Mixtures using column-vector training data provided in \( \text{TrainingData} \).

   If \( \text{DiagonalCovar} \) is true, the components of \( \text{TrainingData}(:,n) \) are assumed to be independent and off diagonals are set to 0.

   Codebook is an optional argument. If it is present, it contains column oriented codewords with the same number of codewords as Mixtures. When present, these codewords are used rather than generating a codebook.

   \( \text{gmm} \) is a structure with the following fields:
   - \( c \) - Vector containing weights of each Gaussian
   - \( \text{means} \) - a cell array where \( \text{means}\{k\} \) is a column vector mean of the \( k \)'th normal distribution.
   - \( \text{cov} \) - a cell array where \( \text{cov}\{k\} \) is the variance-covariance matrix of the \( k \)'th normal distribution.
   - \( \text{DiagonalCovar} \) - true if this GMM is restricted to having diagonal variance-covariance matrices, false otherwise.
The function diag() is useful for the DiagonalCovar case. To diagonalize a matrix, calling it once will return the diagonal elements as a vector. A second call with a diagonal vector will create a full matrix with 0s on the off diagonals.

3. (40 points) Write a function that computes the expectation step for a GMM with the following signature that assumes a model with the same structure as produced by gmmInit:

```matlab
gamma = gmmExpectation(Model, TrainingData)
```

Given an initialized GMM Model and column oriented training data, compute the expectation step of the EM training algorithm as applied to GMMs for the given training data. Matrix gamma should have as many rows as there are mixtures and one column for every training vector.

NOTES ON THE PREVIOUS THREE PROBLEMS:

The blackboard assignment contains function gmmLikelihood.m that computes the probability of a set of feature vectors given a GMM. An optional second output returns the contribution of each mixture to the probability. This will be useful in problem 3. Make sure that you understand the code.

To prevent cascade failure should you have any problems with problems 1 and 2 and to help you in debugging, the Blackboard file gmmtest.mat contains the following to help you:

- features - training data from the TIDIGITS digit one from your last assignment.
- codewords – An example of 4-means clustering of features.
- gmmone – sample output from gmmInit called as follows:
  ```matlab
gmmone = gmmInit(4, features, true, codewords);
  ```
- P – sample output from f_gauss created as follows:
  ```matlab
  P = f_gauss(features, gmmone.means{1}, gmmone.cov{1})
  ```

4. (20 points) For the consonants [m,t] from the word mystery, describe the place of articulation, manner of articulation, and voicing for each one.

5. (20 points) Suppose two tones have pitches of 1250 and 2500 Mel. What can we say about the difference in pitch that the average listener will perceive? What is the approximate frequency in Hz of these two tones?