1. (20 points) Bayes decision rule states that the optimal decision to make is the one that maximizes posterior probability: \( \arg \max_{\omega \in \text{classes}} P(\omega \mid X) \) where \( \omega \) is the class label, and \( X \) is the set of evidence that has been observed. Use Bayes rule to rewrite the posterior probability into a combination of the class-conditional likelihood, prior probability, and probability of observing the evidence (show your work) and label each portion of the resulting equation. In your own words, state what each of these mean. Finally, explain why we frequently omit the probability of observing the evidence when computing Bayes decision rule.

For the speech waveform talking.wav from the previous assignment, modify speech_silence_demo.m to accomplish the goals in the following questions. The function add_normplot is provided to plot a normal distribution on an overlaid axis as shown in class.

2. (20 points) Estimate the mean and variance of the speech distribution using training data between 1.3 and 2.8 s.

3. (60 points) Calculate (in your code) the point at which the speech and noise distributions intersect and display it using fprintf (hint: set the two pdfs to be equal and find the roots of \( x \). While this must be done within your code, do not use Matlab’s symbolic solver, without care the rounding errors will make the result inaccurate. Either use the quadratic equation\(^1\) or Matlab’s root solver (figure out on your own, do not use Matlab’s symbolic solver, without transforming this problem it will not provide a very accurate solution). Plot the following figures:
   a. Histograms of the speech and noise distributions with the noise and speech pdfs overlaid. Use the histogram or hist function Draw a line (plot or line command) on the histograms at the point where the two distributions intersect (must be driven by your calculation, do not insert one by hand).
   b. Plot the original time-domain samples of speech showing which portions are speech and which are silence according to your classifier.

Each figure must have appropriate axis labels. Write a caption to accompany each plot. Good captions are important in scientific papers as readers frequently thumb through a publication looking at the figures before they read the article. Hence, the caption should contain enough information to give a reader familiar with the field a sense of what the figure represents. This typically requires one or two sentences.

\(^1\) Recall from high school that the quadratic equation, \( \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \), finds the roots for polynomial \( ax^2 + bx + c \) .
Write the following functions related to K-Means clustering. These functions should work on multidimensional data consisting of column vectors organized into a matrix. As an example, the matrix $\begin{bmatrix} 1.5 & 1.4 & 1.5 & 0.8 \\ 2.3 & 4.2 & 4.5 & 4.1 \\ 2.6 & 3.3 & 1.8 & 3.0 \end{bmatrix}$ represents four observations in $\mathbb{R}^3$ space:

$\begin{bmatrix} 1.5 \\ 2.3 \\ 2.6 \end{bmatrix}$, $\begin{bmatrix} 1.4 \\ 4.2 \\ 3.3 \end{bmatrix}$, $\begin{bmatrix} 1.5 \\ 4.5 \\ 1.8 \end{bmatrix}$, and $\begin{bmatrix} 0.8 \\ 4.1 \\ 3.0 \end{bmatrix}$. Your functions should support $\mathbb{R}^D$ where $D \geq 1$.

4. (10 points) Write a function that selects $k$ initial codewords (means, or centroids) from a data matrix at random:
   
   ```matlab```
   function codewords = kmeansinit(k, Matrix)
   if have the statistics toolbox, the function randsamp will be useful. Otherwise,
   use the function randperm and pick the first k samples
   return
   end
   codewords
   ```
   (the initial means in the feature space).

5. (20 points) Write the following Matlab function:
   ```matlab```
   function [d, mincol] = eucliddist(Vector, Matrix)
   returns row vector d, containing the Euclidian distortion between the
   given column Vector and each column of Matrix. You may need to use a
   loop for this function, although if you are adventurous you can read about
   bsxfun which will let you solve this without a loop over the column
   vectors. Vector mincol should tell you which column of Matrix produced
   the minimum distortion. Read about the min function’s output arguments to
   make this part easy.
   ```
   Concrete example:
   ```matlab```
   [d, bestidx] = eucliddist([3;4], [3 4 5; 4 5 6])
   ```
   returns $d = [0 2 8]$ and bestidx = 1.

What to turn in:

- Hard copy of everything.
- Submit Matlab code to Blackboard (still need a hard copy)