MUSI 6201
Assignment 3: Genre Classification
Group: Liang Tang, Chris Latina

Q1 Covariance matrix visualization

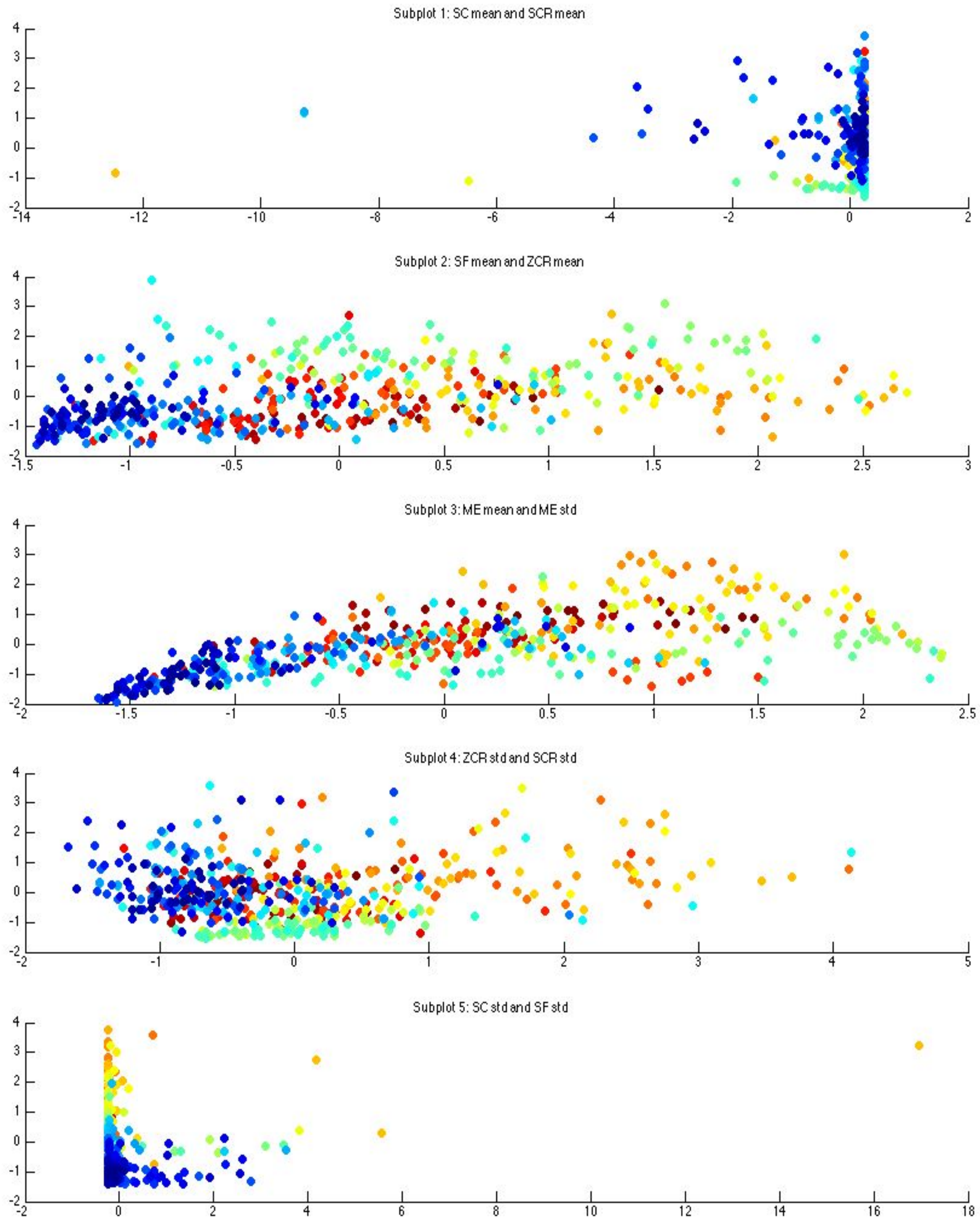
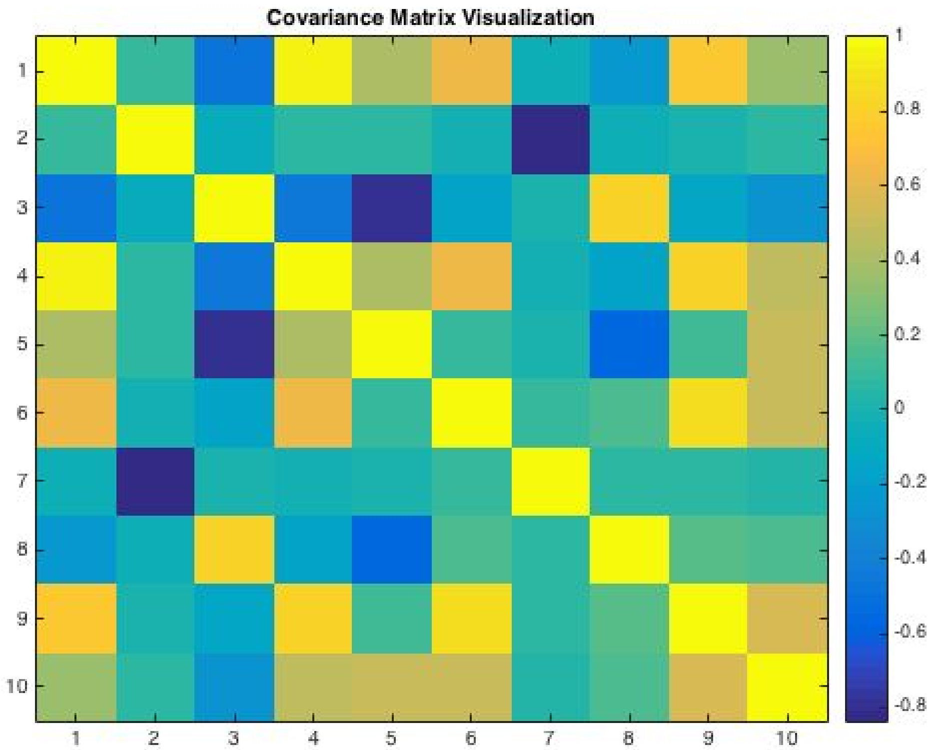


Table of Features

1	Mean of Maximum Envelope	6	Standard Deviation of Maximum Envelope
2	Mean of Spectral Centroid	7	Standard Deviation of Spectral Centroid
3	Mean of Spectral Crest	8	Standard Deviation of Spectral Crest
4	Mean of Spectral Flux	9	Standard Deviation of Spectral Flux
5	Mean of Zero Crossing Rate	10	Standard Deviation of Zero Crossing Rate



This visualization represents how much two features change together. The figure above, Covariance Matrix Visualization, the values on the diagonal equal 1: each feature always changes with itself perfectly. Also, it is obvious that all the color blocks are symmetrical about the diagonal. For example, the block at (7,2) has the same color as the block at (2,7), since covariance is order independent. Looking more closely, the block at (5,3) has a relatively low value which can be explained by the low correlation between the mean of zero crossing and mean of spectral crest.

Q2 Solution to the problem of equal distance: our test and training data are 50x10 and 450x10 respectively. The distance vector represents the comparision of each instance, reducing the 10-deature dimension to a single distance value.

```
for i=1:num_test
   for j=1:num_train
        distance(i,j) = sqrt(sum ((testData(i,:) - trainData(j,:)).^2));
    end
end
```

We then sort all distances and select the smallest K distances. We then index into the genre labels to associate each distance with the correct genre class. After counting these, we select the genre with the maximum number of occurances. If there is a tie, we take the occurance with the smallest cumulative distance.

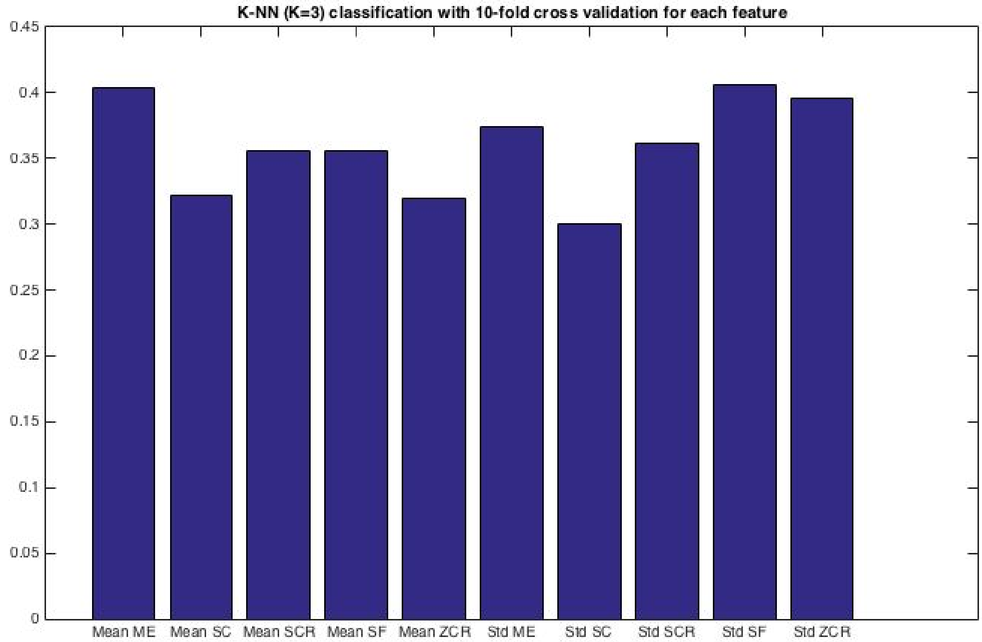
```
% Take the majority with the smallest cumulative distance
for j = 1: size(I,1)
    counts = zeros(num_labels,K);
    sums = zeros(num_labels,1);
    for i = 1:num_labels
        counts(i,:) = strcmp(all_labels(i),genres(j,:));
        sums(i,:) = sum(counts(i,:).*k_dist(j,:));
    end

    num_matches = sum(counts,2);

    % find ind where num of matches is max
    maxes = (sum(counts,2) == max(sum(counts,2)));
    max_labels = maxes.*sums;

    %Take the min distance
    min_dist_inds = find(max_labels);
    if isempty(min_dist_inds)
        %estimatedClass(j,1) = genres(j,1);
        min_dist_inds = 1;
    end
    [min_dist, min_dist_ind] = min(max_labels(min_dist_inds));
    estimatedClass(j,1) = all_labels(min_dist_inds(min_dist_ind));
end
```

Q3

1. *Rank the single best feature using 10-fold cross validation.*

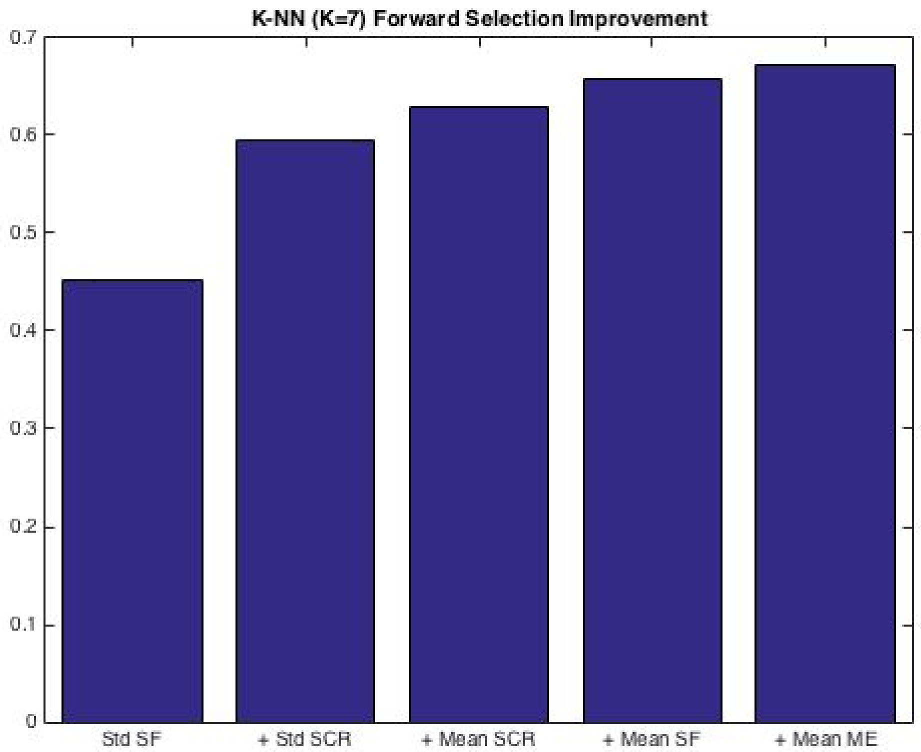
   The following rates from left to right are:
   [0.404,0.322,0.356,0.356,0.320,0.374,0.300,0.362,0.406,0.396].

   Standard deviation of spectral flux, mean max envelope, and standard deviation of spectral crest were the top three features.

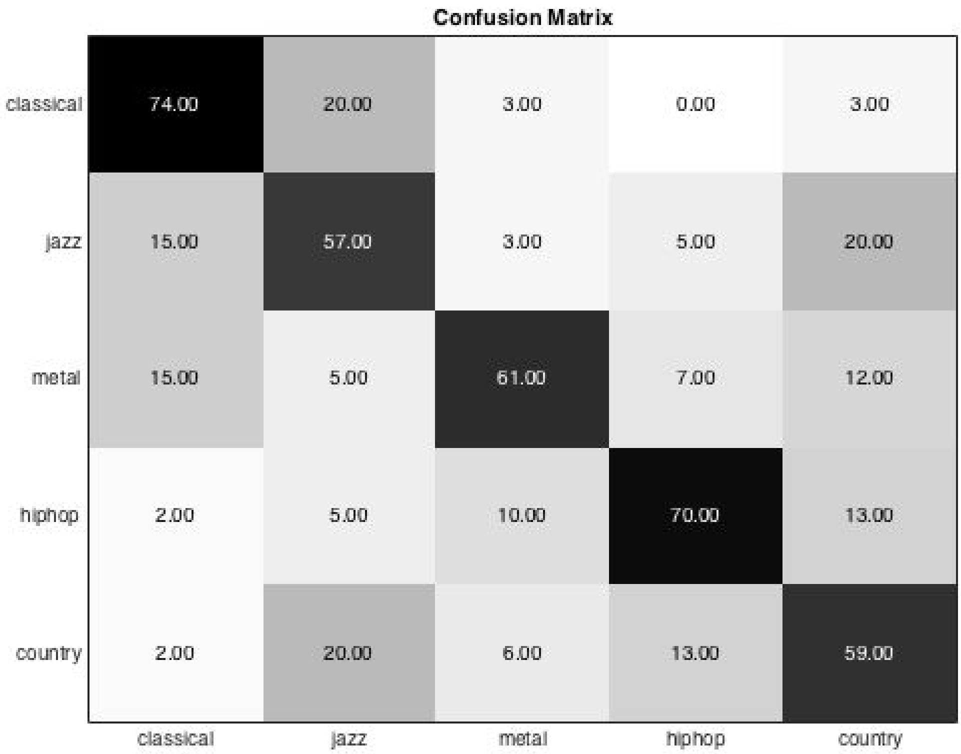


2. *Implement your Sequential Forward Selection based on the performance of 10-fold cross validation. Plot your classification performance depending on the number of features selected.*

   The chart below shows the performance improvement at each step of the recursive Sequential Forward Selection algorithm with K = 7. Our features selected were [Std. SF, Std SCR, Mean SCR, Mean SF, Mean ME] with stepRate = [0.4520, 0.5940,0.6300, 0.6580, 0.6720].

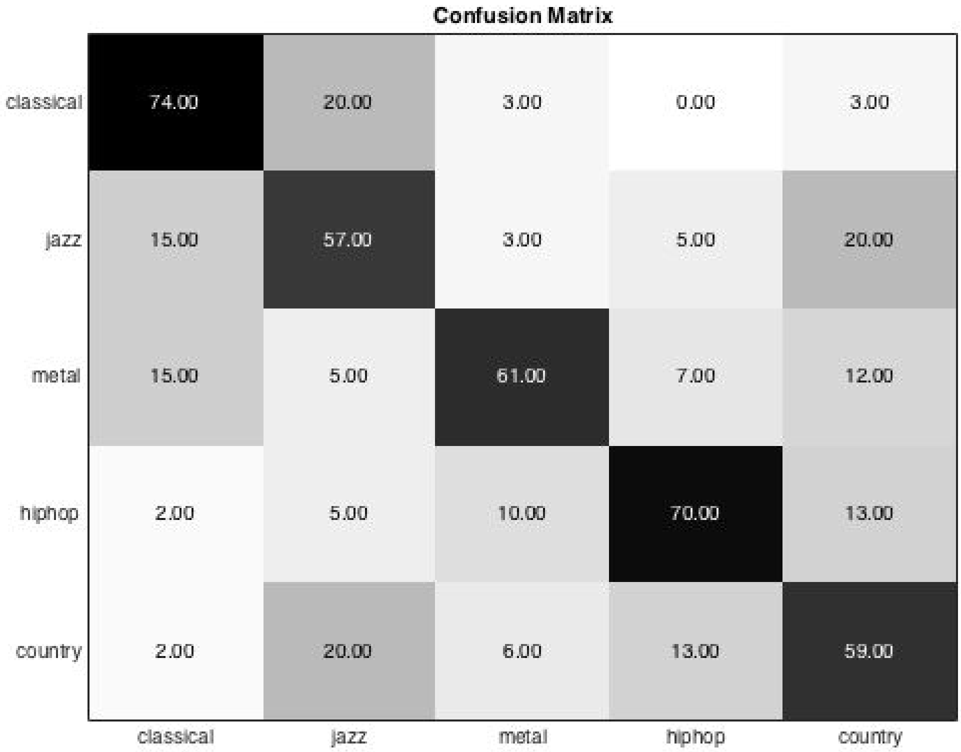


3. Confusion Matrix visualization.
    ○ **K = 1**, Features = [9,1,4,2,3,8,10,7,5,6] , Mean Rate = **0.6420**



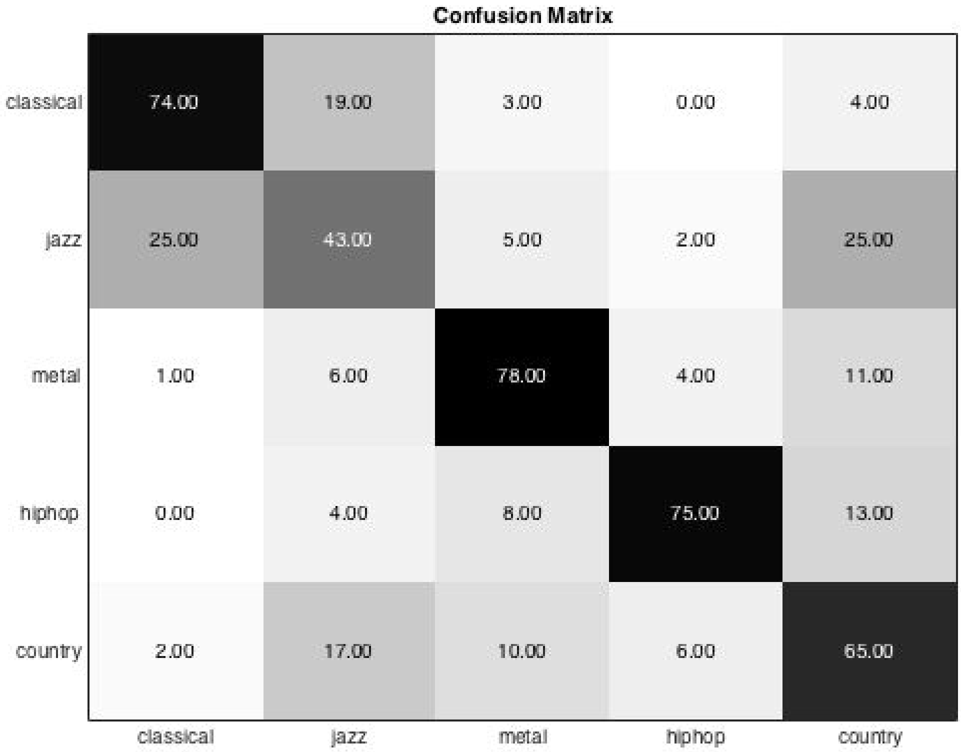
With K=1, all of the features were selected in the sequential foward selection. There is a lot of confusion over jazz and country and jazz and classical music. It appeats that hiphop and classical are easily distinguishable.

- **K = 3**, Features = [9,3,4,10,5,7,1,8,2], Mean Rate = **0.6800**



With K = 3, almost all of the features were used! 9 of them! Again, hiphop and classical were easily distinguishable. The results were also better than when K=1 with a mean rate of 0.6800 correct matches. The confusion between country and jazz and between jazz and classical is still quite significant.

- **K = 7**, Features = [9,8,3,4,1], Mean Rate = **0.6720**



The results for K = 7 was slightly worse than K=3, however fewer features were required and the match for metal was the highest rating of them all at 78% correct. There is a significantly low rate of matches for jazz with this configuration, as it is often confused again with country and classical. With all K values, the first feature selected was the same, namely number 9: standard deviation of spectral flux. I think that some **higher level musical features** using MFCCs and pitch chroma, onset density, etc would definitely help differentiate these genres.

Note: It's best to run the different portions of evaluateResults individually so you can see the output of the forward selection, rates, etc in the console and Workspace variables. We also have a Main.m function you can use to test our functions.