CS 529 – Assignment 3 - Logistic Regression

Vamshi Krishna N S

1. Introduction

In this assignment, I worked on classifying the songs according to the music genres such as classical, jazz, country, pop, rock and metal.

I have programmed in "Python 2.7.9 |Anaconda 2.2.0 (64-bit)|" installed in Windows 8.1.

Program Execution:

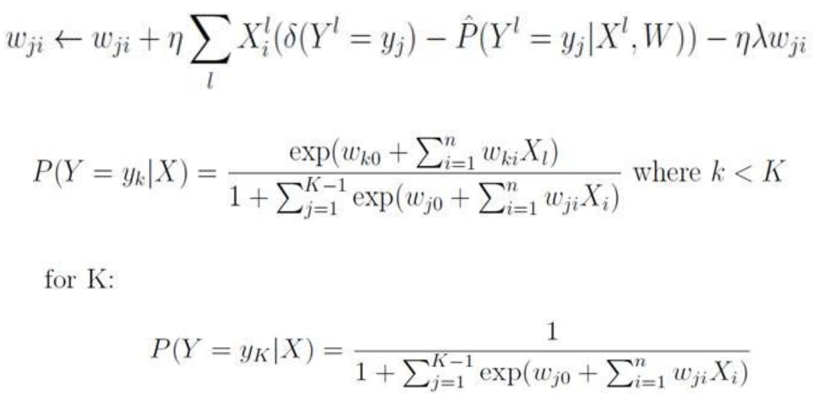
LogisticRegression.py <option><path to data>

<option>	-fft	Generates fft for the data and performs the Logistic Regression with Gradient Ascent
	-fft20	Generates 20 best features per genre for the fft data and performs the Logistic Regression with Gradient Ascent
	-mfcc	Generates mfcc for the data and performs the Logistic Regression with Gradient Ascent
<path to data>	Path to the music data. Eg: <HOME_DIR>\opihi.cs.uvic.ca\sound\genres	

2. Procedure Followed:

I have used Logistic Regression as the classifier (as the data contains non-boolean classes) which uses Gradient Descent to calculate its weights.

The following equations from Linear Regression by Tom Mitchell were very helpful in writing the code.



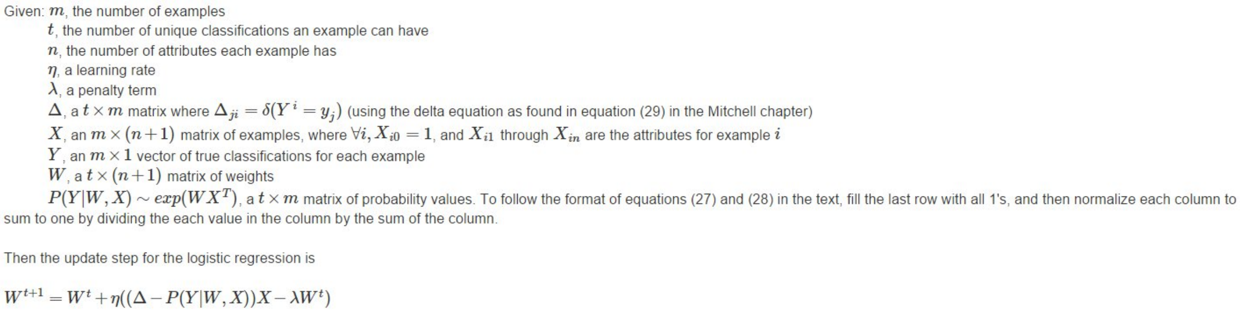
$$w_{ji} \leftarrow w_{ji} + \eta \sum_{l} X_i^l(\delta(Y^l = y_j) - \hat{P}(Y^l = y_j|X^l, W)) - \eta\lambda w_{ji}$$

$$P(Y = y_k|X) = \frac{\exp(w_{k0} + \sum_{i=1}^{n} w_{ki}X_l)}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{n} w_{ji}X_i)} \text{ where } k < K$$

for K:

$$P(Y = y_K|X) = \frac{1}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{n} w_{ji}X_i)}$$

Matrix formulation:



Given: m, the number of examples

t, the number of unique classifications an example can have

n, the number of attributes each example has

η, a learning rate

λ, a penalty term

Δ, a $t \times m$ matrix where $\Delta_{ji} = \delta(Y^i = y_j)$ (using the delta equation as found in equation (29) in the Mitchell chapter)

X, an $m \times (n+1)$ matrix of examples, where $\forall i, X_{i0} = 1$, and X_{i1} through X_{in} are the attributes for example i

Y, an $m \times 1$ vector of true classifications for each example

W, a $t \times (n+1)$ matrix of weights

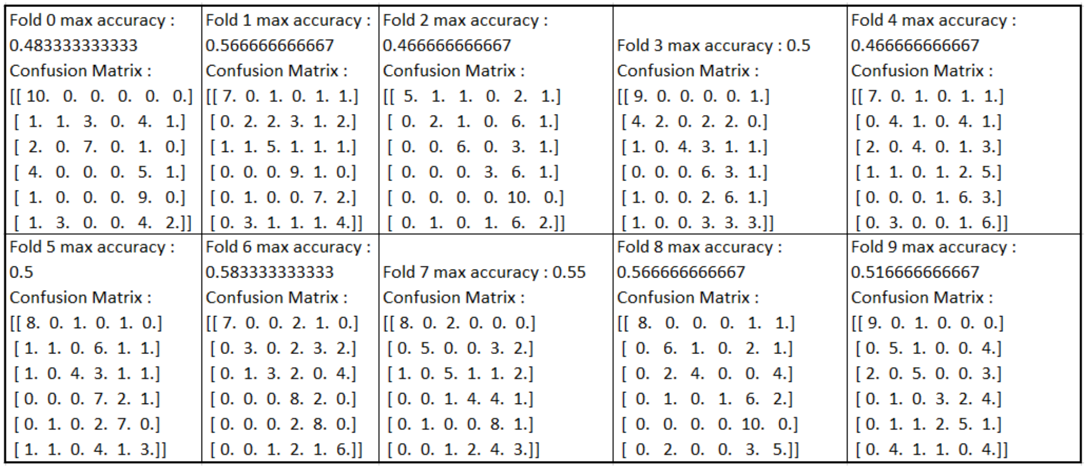
$P(Y|W,X) \sim exp(WX^T)$, a $t \times m$ matrix of probability values. To follow the format of equations (27) and (28) in the text, fill the last row with all 1's, and then normalize each column to sum to one by dividing the each value in the column by the sum of the column.

Then the update step for the logistic regression is

$$W^{t+1} = W^t + \eta((\Delta - P(Y|W,X))X - \lambda W^t)$$

A) Use the 1000 first FFT components as features.

- ✓ I have generated the FFT features from the data provided and formulated a matrix of size (600x1000).
- ✓ Then separated the data into train and test matrices using k-fold cross validation where k = 10.
- ✓ Performed normalization on features and not on the entire data which removes the bias on certain features.
- ✓ Performed Logistic Regression using Gradient Descent method for 300 iterations.
- ✓ I have got an average accuracy rate of 52% for all the 10 folds data.
- ✓ Following are the Accuracies and Confusion Matrices for all 10 folds data.
  Confusion Matrix follows the order: 'classical', 'country', 'jazz', 'metal', 'pop', 'rock'
  (Actual vs Classification)



Fold 0 max accuracy : 0.483333333333 Confusion Matrix : [[10. 0. 0. 0. 0. 0.] [1. 1. 3. 0. 4. 1.] [2. 0. 7. 0. 1. 0.] [4. 0. 0. 0. 5. 1.] [1. 0. 0. 0. 9. 0.] [1. 3. 0. 0. 4. 2.]]	Fold 1 max accuracy : 0.566666666667 Confusion Matrix : [[7. 0. 1. 0. 1. 1.] [0. 2. 2. 3. 1. 2.] [1. 1. 5. 1. 1. 1.] [0. 0. 0. 9. 1. 0.] [0. 1. 0. 0. 7. 2.] [0. 3. 1. 1. 1. 4.]]	Fold 2 max accuracy : 0.466666666667 Confusion Matrix : [[5. 1. 1. 0. 2. 1.] [0. 2. 1. 0. 6. 1.] [0. 0. 6. 0. 3. 1.] [0. 0. 0. 3. 6. 1.] [0. 0. 0. 0. 10. 0.] [0. 1. 0. 1. 6. 2.]]	Fold 3 max accuracy : 0.5 Confusion Matrix : [[9. 0. 0. 0. 0. 1.] [4. 2. 0. 2. 2. 0.] [1. 0. 4. 3. 1. 1.] [0. 0. 0. 6. 3. 1.] [1. 0. 0. 2. 6. 1.] [1. 0. 0. 3. 3. 3.]]	Fold 4 max accuracy : 0.466666666667 Confusion Matrix : [[7. 0. 1. 0. 1. 1.] [0. 4. 1. 0. 4. 1.] [2. 0. 4. 0. 1. 3.] [1. 1. 0. 1. 2. 5.] [0. 0. 0. 1. 6. 3.] [0. 3. 0. 0. 1. 6.]]
Fold 5 max accuracy : 0.5 Confusion Matrix : [[8. 0. 1. 0. 1. 0.] [1. 1. 0. 6. 1. 1.] [1. 0. 4. 3. 1. 1.] [0. 0. 0. 7. 2. 1.] [0. 1. 0. 2. 7. 0.] [1. 1. 0. 4. 1. 3.]]	Fold 6 max accuracy : 0.583333333333 Confusion Matrix : [[7. 0. 0. 2. 1. 0.] [0. 3. 0. 2. 3. 2.] [0. 1. 3. 2. 0. 4.] [0. 0. 0. 8. 2. 0.] [0. 0. 0. 2. 8. 0.] [0. 0. 1. 2. 1. 6.]]	Fold 7 max accuracy : 0.55 Confusion Matrix : [[8. 0. 2. 0. 0. 0.] [0. 5. 0. 0. 3. 2.] [1. 0. 5. 1. 1. 2.] [0. 0. 1. 4. 4. 1.] [0. 1. 0. 0. 8. 1.] [0. 0. 1. 2. 4. 3.]]	Fold 8 max accuracy : 0.566666666667 Confusion Matrix : [[8. 0. 0. 0. 1. 1.] [0. 6. 1. 0. 2. 1.] [0. 2. 4. 0. 0. 4.] [0. 1. 0. 1. 6. 2.] [0. 0. 0. 0. 10. 0.] [0. 2. 0. 0. 3. 5.]]	Fold 9 max accuracy : 0.516666666667 Confusion Matrix : [[9. 0. 1. 0. 0. 0.] [0. 5. 1. 0. 0. 4.] [2. 0. 5. 0. 0. 3.] [0. 1. 0. 3. 2. 4.] [0. 1. 1. 2. 5. 1.] [0. 4. 1. 1. 0. 4.]]

- ✓ Avg of all folds accuracies : 0.52 (52%)
- ✓ From the confusion matrices above, ‘metal, ‘pop’, ’rock’ are similar to each other.
- ✓ Thismisclassification may be due to the similar frequencies in the songs which may be due to the use of similar instruments. Due to this some of the songs may have become biased towards other genres.

B) Using your knowledge from the previous homework, design a method to rank the FFT components and select the best 20 per genre. Use the selected 120 features to classify the data set. Explain how this step affects your accuracy.

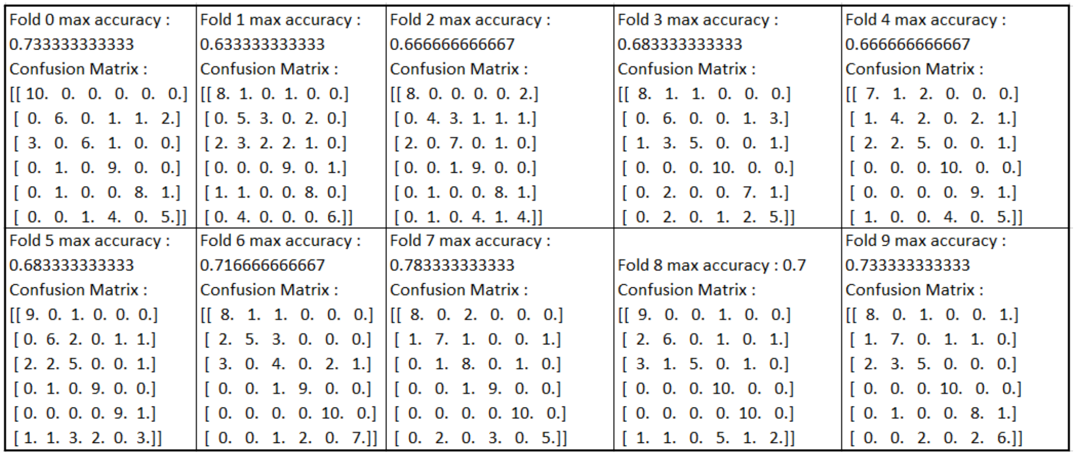
- ✓ I have generated the FFT features from the data provided and formulated a matrix of size (600x1000).
- ✓ Select a genre and compute Standard Deviation of the features in the data per genre (100x1000). This forms matrix of size (1x1000).
- ✓ Compute Standard Deviation of the features for entire data (600x1000). This forms matrix of size (1x1000).
- ✓ Compute the difference between the two resultant matrices.
- ✓ Select the indexes of top 20 Standard Deviation values from the matrix in the previous step.
- ✓ For the selected indexes, extract the data from those 20 features. So the new matrix formed is of the size (100x20).
- ✓ Repeat these steps for all the genres.
- ✓ Append all the matrices using “np.r_” which forms matrix of size (600x20) which constitutes of all the genre data across 120 features.
- ✓ For the generated data, performed Logistic Regression using Gradient Descent for 300 iterations.
- ✓ Following are the Accuracies and Confusion Matrices for all 10 folds data.
  Confusion Matrix follows the order: 'classical', 'country', 'jazz', 'metal', 'pop', 'rock'
  (Actual vs Classification)

Fold 0 max accuracy : 0.633333333333 Confusion Matrix : [[9. 0. 0. 0. 1. 0.] [2. 5. 2. 1. 0. 0.] [2. 2. 5. 0. 1. 0.] [1. 0. 0. 9. 0. 0.] [0. 1. 2. 2. 5. 0.] [2. 1. 0. 1. 1. 5.]]	Fold 1 max accuracy : 0.616666666667 Confusion Matrix : [[7. 1. 2. 0. 0. 0.] [3. 5. 2. 0. 0. 0.] [0. 4. 6. 0. 0. 0.] [0. 0. 1. 9. 0. 0.] [1. 2. 2. 0. 5. 0.] [0. 3. 1. 1. 0. 5.]]	Fold 2 max accuracy : 0.6 Confusion Matrix : [[7. 2. 0. 0. 0. 1.] [0. 6. 1. 0. 3. 0.] [1. 1. 7. 0. 1. 0.] [1. 0. 2. 5. 2. 0.] [0. 2. 1. 0. 5. 2.] [1. 1. 0. 2. 0. 6.]]	Fold 3 max accuracy : 0.6 Confusion Matrix : [[8. 0. 1. 0. 1. 0.] [4. 4. 1. 0. 1. 0.] [1. 3. 4. 0. 2. 0.] [1. 1. 3. 5. 0. 0.] [0. 0. 1. 0. 9. 0.] [1. 1. 1. 0. 1. 6.]]	Fold 4 max accuracy : 0.583333333333 Confusion Matrix : [[9. 0. 1. 0. 0. 0.] [0. 6. 3. 0. 1. 0.] [3. 2. 4. 0. 0. 1.] [2. 0. 3. 5. 0. 0.] [1. 0. 5. 0. 3. 1.] [1. 0. 1. 0. 0. 8.]]
Fold 5 max accuracy : 0.616666666667 Confusion Matrix : [[8. 0. 2. 0. 0. 0.] [1. 4. 2. 2. 1. 0.] [1. 1. 7. 1. 0. 0.] [2. 0. 1. 7. 0. 0.] [1. 1. 3. 0. 4. 1.] [3. 0. 0. 0. 0. 7.]]	Fold 6 max accuracy : 0.616666666667 Confusion Matrix : [[6. 0. 4. 0. 0. 0.] [3. 6. 1. 0. 0. 0.] [2. 2. 5. 1. 0. 0.] [1. 0. 4. 5. 0. 0.] [0. 1. 2. 0. 7. 0.] [0. 0. 2. 0. 0. 8.]]	Fold 7 max accuracy : 0.633333333333 Confusion Matrix : [[8. 0. 1. 0. 1. 0.] [0. 6. 2. 0. 2. 0.] [3. 0. 6. 0. 1. 0.] [0. 2. 2. 6. 0. 0.] [1. 1. 3. 0. 5. 0.] [1. 0. 0. 0. 2. 7.]]	Fold 8 max accuracy : 0.666666666667 Confusion Matrix : [[8. 0. 2. 0. 0. 0.] [2. 5. 3. 0. 0. 0.] [1. 0. 8. 0. 1. 0.] [1. 0. 2. 7. 0. 0.] [0. 0. 4. 0. 6. 0.] [1. 1. 2. 0. 0. 6.]]	Fold 9 max accuracy : 0.666666666667 Confusion Matrix : [[8. 0. 1. 0. 0. 1.] [0. 5. 5. 0. 0. 0.] [0. 1. 8. 0. 1. 0.] [0. 1. 0. 8. 1. 0.] [0. 2. 2. 0. 5. 1.] [0. 2. 2. 0. 0. 6.]]

- ✓ Avg of all folds accuracies :0.623 (62.3%)
- ✓ This increased average accuracy around 10% although 880 features are removed from the data.
- ✓ Some of the songs in Jazz are classified into classical and country. Also Jazz and Pop are similarly classified.
- ✓ This misclassification may be due to the selection of similar features during the selection of best 20 features across the genres. Due to this some of the songs may have become biased towards other genres.

C)Using MFCC features.

- ✓ Extracting features using MFCC took less time than FFT. Also it has 13 features compared to 1000 features of FFT. This made the computations lesser and thereby decreasing the execution time.
- ✓ Then separated the data into train and test matrices using k-fold cross validation where k = 10.
- ✓ Performed normalization on features and not on the entire data which removes the bias on certain features.
- ✓ Performed Logistic Regression using Gradient Descent method for 300 iterations.
- ✓ I have got an average accuracy rate of 52% for all the 10 folds data.
- ✓ Following are the Accuracies and Confusion Matrices for all 10 folds data.
  Confusion Matrix follows the order: 'classical', 'country', 'jazz', 'metal', 'pop', 'rock'
  (Actual vs Classification)



Fold 0 max accuracy : 0.733333333333 Confusion Matrix : [[10. 0. 0. 0. 0. 0.] [0. 6. 0. 1. 1. 2.] [3. 0. 6. 1. 0. 0.] [0. 1. 0. 9. 0. 0.] [0. 1. 0. 0. 8. 1.] [0. 0. 1. 4. 0. 5.]]	Fold 1 max accuracy : 0.633333333333 Confusion Matrix : [[8. 1. 0. 1. 0. 0.] [0. 5. 3. 0. 2. 0.] [2. 3. 2. 2. 1. 0.] [0. 0. 0. 9. 0. 1.] [1. 1. 0. 0. 8. 0.] [0. 4. 0. 0. 0. 6.]]	Fold 2 max accuracy : 0.666666666667 Confusion Matrix : [[8. 0. 0. 0. 0. 2.] [0. 4. 3. 1. 1. 1.] [2. 0. 7. 0. 1. 0.] [0. 0. 1. 9. 0. 0.] [0. 1. 0. 0. 8. 1.] [0. 1. 0. 4. 1. 4.]]	Fold 3 max accuracy : 0.683333333333 Confusion Matrix : [[8. 1. 1. 0. 0. 0.] [0. 6. 0. 0. 1. 3.] [1. 3. 5. 0. 0. 1.] [0. 0. 0. 10. 0. 0.] [0. 2. 0. 0. 7. 1.] [0. 2. 0. 1. 2. 5.]]	Fold 4 max accuracy : 0.666666666667 Confusion Matrix : [[7. 1. 2. 0. 0. 0.] [1. 4. 2. 0. 2. 1.] [2. 2. 5. 0. 0. 1.] [0. 0. 0. 10. 0. 0.] [0. 0. 0. 0. 9. 1.] [1. 0. 0. 4. 0. 5.]]
Fold 5 max accuracy : 0.683333333333 Confusion Matrix : [[9. 0. 1. 0. 0. 0.] [0. 6. 2. 0. 1. 1.] [2. 2. 5. 0. 0. 1.] [0. 1. 0. 9. 0. 0.] [0. 0. 0. 0. 9. 1.] [1. 1. 3. 2. 0. 3.]]	Fold 6 max accuracy : 0.716666666667 Confusion Matrix : [[8. 1. 1. 0. 0. 0.] [2. 5. 3. 0. 0. 0.] [3. 0. 4. 0. 2. 1.] [0. 0. 1. 9. 0. 0.] [0. 0. 0. 0. 10. 0.] [0. 0. 1. 2. 0. 7.]]	Fold 7 max accuracy : 0.783333333333 Confusion Matrix : [[8. 0. 2. 0. 0. 0.] [1. 7. 1. 0. 0. 1.] [0. 1. 8. 0. 1. 0.] [0. 0. 1. 9. 0. 0.] [0. 0. 0. 0. 10. 0.] [0. 2. 0. 3. 0. 5.]]	Fold 8 max accuracy : 0.7 Confusion Matrix : [[9. 0. 0. 1. 0. 0.] [2. 6. 0. 1. 0. 1.] [3. 1. 5. 0. 1. 0.] [0. 0. 0. 10. 0. 0.] [0. 0. 0. 0. 10. 0.] [1. 1. 0. 5. 1. 2.]]	Fold 9 max accuracy : 0.733333333333 Confusion Matrix : [[8. 0. 1. 0. 0. 1.] [1. 7. 0. 1. 1. 0.] [2. 3. 5. 0. 0. 0.] [0. 0. 0. 10. 0. 0.] [0. 1. 0. 0. 8. 1.] [0. 0. 2. 0. 2. 6.]]

- ✓ Avg of all folds accuracies : 0.70 (70%)
- ✓ In MFCC –
  - o Jazz, country are similar to each other.
  - o Metal, rock are similar to each other

3. Conclusions

- ✓ FFT didn’t produce high accuracy as MFCC
- ✓ But selecting top 20 features per genre increased the accuracy rate.
- ✓ MFCC produced high accuracy results and the execution time is very less compared to FFT.
- ✓ This classification can further be improved by taking more features into consideration, although it increases the execution time, to improve the training weights and thereby increase accuracy.
- ✓ Another way it can be improved is by using Logistic Regression over Bernouli Distribution, where Y is a Boolean. This means that, the Logistic Regression method checks whether the song belongs to a particular genre or not. This check is done against all the genres in the experiment. Then take the argmax of the Boolean values, the index at which gives the genre of a particular song.

4. References

[1] Tom M. Mitchell, Machine Learning (1) [pdf]. Available: http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf