Santosh Yadav Jada

Student Id: 16245773

Mini-project #1: Fingerprint Spoof Detector using Bayes Classifier

Aim: The aim of this project is to design a spoof detector using Naïve Bayes Classifier.

Software Used: MATLAB 2017

Procedure:

1. Load the data (.mat files).
2. Add all training data.
3. Create labels (response variable).
4. Use fitcnb function to generate a Naïve Bayes Model for the training data and also set prior.
5. Add all the testing data and run the model with this to get the output.
6. Use loss function and resubLoss function to get respective values.

Code:

a) Liv vs Latex:

```
load('featureMat_Liv_train_bioLBP')
load('featureMat_Latex_train_bioLBP')
load('featureMat_Liv_test_bioLBP')
load('featureMat_Latex_test_bioLBP')

train_data=[featureMat_liv_train_bioLBP'
featureMat_Latex_train_bioLBP'];
labels=cat(2,ones(1,1000),zeros(1,200));
NBmodel=fitcnb(train_data',labels','Prior',[0.5 0.5]);
NBmodel.Prior

%change prior to [0.6 0.4]

test_data=[featureMat_liv_test_bioLBP' featureMat_Latex_test_bioLBP'];
test_output = predict(NBmodel,test_data');

LossLat = loss(NBmodel, test_data', labels')

ResubLat = resubLoss(NBmodel)
```

b) Liv Vs Gelatine:

```
load('featureMat_Liv_train_bioLBP')
load('featureMat_Gelatine_train_bioLBP')

load('featureMat_Liv_test_bioLBP')
load('featureMat_Gelatine_test_bioLBP')

train_data=[featureMat_liv_train_bioLBP'
featureMat_Gelatine_train_bioLBP'];
labels=cat(2,ones(1,1000),zeros(1,200));
NBmodel=fitcnb(train_data',labels','Prior',[0.5 0.5]);
NBmodel.Prior
%Change NBmodel.Prior = [0.6 0.4]

test_data=[featureMat_liv_test_bioLBP'
featureMat_Gelatine_test_bioLBP'];
test_output = predict(NBmodel,test_data');


LossGela = loss(NBmodel, test_data', labels')

ResubGela = resubLoss(NBmodel)
```

Observations:

LivVsLatex	Prior[0.5 0.5]	Prior[0.6 0.4]	Default Prior
Loss	0.3265	0.3606	0.2008
Resubloss	0.1065	0.1036	0.1000

LivVsGelatine	Prior[0.5 0.5]	Prior[0.6 0.4]	Default Prior
Loss	0.2185	0.1986	0.2792
Resubloss	0.1965	0.1922	0.1983

Comments:

Loss is the cost of the performance task when prediction is done by the model. If the model yields smaller loss values then it is better.

Re-substitution Loss: The re-substitution loss is the loss between the response training data and the model's predicted response values based on input training data. If re-substitution value is high then the model's predictions are not good and if it's too low then it doesn't guarantee good predictions for new data.

From the observations, we could see that when we have default prior then the model created using Naïve Bayes has a loss=0.2008 with the Latex finger prints and 0.2792 with Gelatine prints. The resub values are 0.1 for Latex and 0.1983 for the Gelatine. So, the model with the Gelatine prints is better than the Latex for default prior. If we look at the values for different priors, we could infer that the model with the Gelatine prints is better than the Latex one as a fingerprint spoof detector.