Manotej Meka
CSE 491 – Biometrics

Project 1 Report

1) Genuine and Impostor Scores

Count	Genuine	Impostor
Finger Print	450	450
Hand	450	450

2) Maximum and Minimum Score

Scores	Maximum	Minimum
Finger Print	966	0
Hand	626	0

3) Mean and Variance

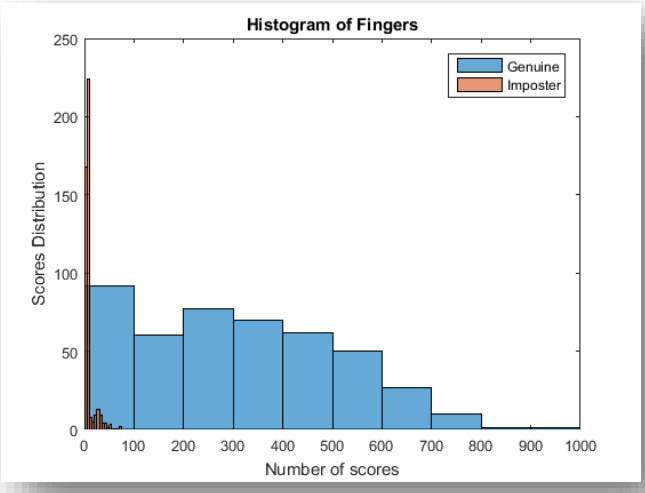
Finger Print	Mean	Variance
Genuine	306.58	40916
Imposter	7.9711	91.008

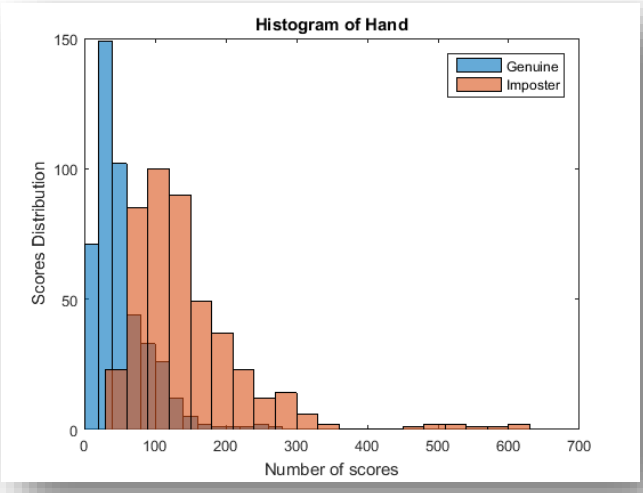
Hand	Mean	Variance
Genuine	50.644	1519.7
Imposter	144.44	6941.1

4) D-Prime

Scores	D-Prime
Finger Print	2.0854
Hand	1.442

5) Histograms

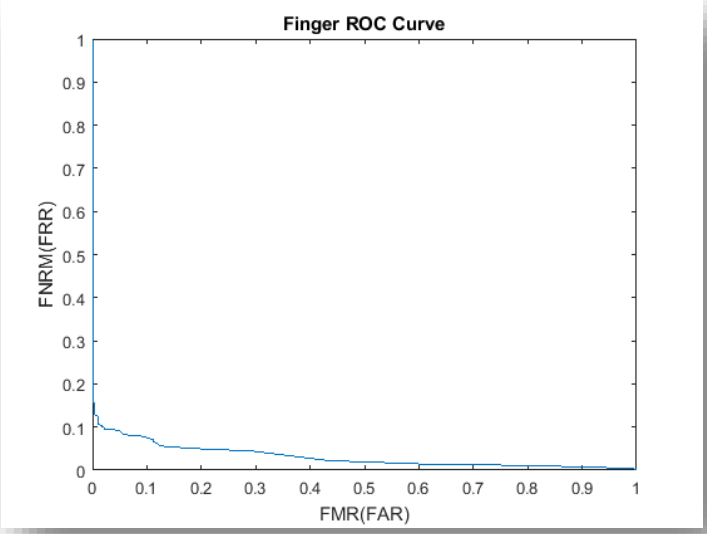


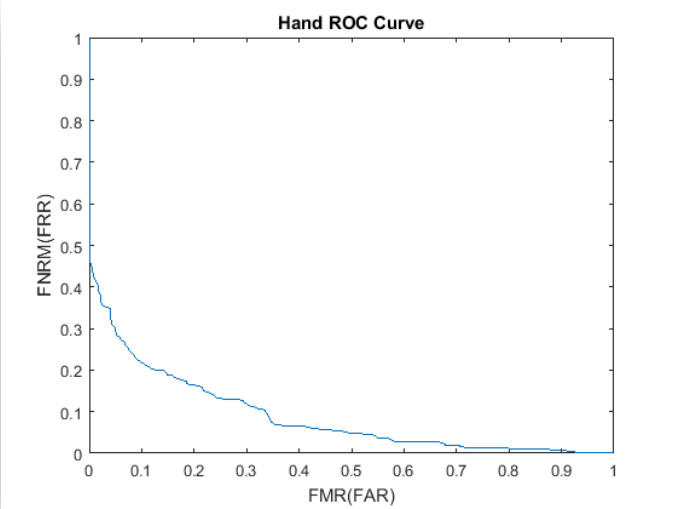


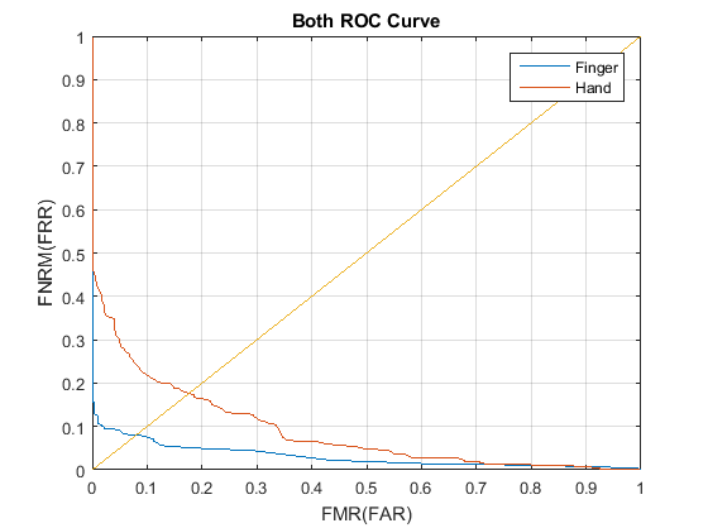
6) FMR and FNMR

Threshold	FMR (*100 for %)	FNMR (*100 for %)
Fingerprint (32)	0.04222 (4.22%)	0.09333 (9.34%)
Hand (45)	0.0044444 (0.45%)	0.44889 (44.89%)

7) ROC and EER







	Finger	Hand
EER @ 45 degrees	8	17.6

8) FNRM @ FMR

FMNR @	FMR 10 %	FMR 1%	FMR 0.1%
Fingerprint	7.56	12.4	15.56
Hand	22.22	42.0	46.45

9) Opinion Matcher
   a. After looking at all of the above statistics and graphs, the fingerprint matcher performed better than the hand matcher. To start of fingerprint matcher has the higher D' value meaning a better match. The ROC for the fingerprint is lower through the various thresholds compared the hand. Lastly, the EER matcher is lower for the finger at 8 instead of 17.6 for the hand.

Project 1 Code

```
clear;
clc;
close all;

% Opening the files
Finger_Gen_File = 'finger_genuine.score';
Finger_Imp_File = 'finger_impostor.score';
Hand_Gen_File = 'hand_genuine.score';
Hand_Imp_File = 'hand_impostor.score';

formatSpec = '%f';

% Reading and extracting files
Gen_Finger_Array = fscanf(fopen(Finger_Gen_File,'r'),formatSpec);
Imp_Finger_Array = fscanf(fopen(Finger_Imp_File,'r'),formatSpec);
Gen_Hand_Array = fscanf(fopen(Hand_Gen_File,'r'),formatSpec);
Imp_Hand_Array = fscanf(fopen(Hand_Imp_File,'r'),formatSpec);
fclose('all');

Gen_Fing_Count = length(Gen_Finger_Array);
Imp_Fing_Count = length(Imp_Finger_Array);
Gen_Hand_Count = length(Gen_Hand_Array);
Imp_Hand_Count = length(Imp_Hand_Array);

fprintf('\tTotal number of Genuine and Imposter scores:\n')
disp(table({'Genuine';'Imposter'},[Gen_Fing_Count;Imp_Fing_Count],[Gen_Hand_Count;Imp_Han
d_Count],'VariableNames',{'TotalCount' 'Finger' 'Hand'}))

%------------------------------------------------------------------------
% Min and Max
Fin_max = max(vertcat(Gen_Finger_Array,Imp_Finger_Array));
Fin_min = min(vertcat(Gen_Finger_Array,Imp_Finger_Array));
Han_max = max(vertcat(Gen_Hand_Array,Imp_Hand_Array));
Han_min = min(vertcat(Gen_Hand_Array,Imp_Hand_Array));

fprintf('\nMax and Min scores generated by each matcher:\n')
disp(table({'Maximum';'Minimum'},[Fin_max;Fin_min],[Han_max;Han_min],'VariableNames',{'Sc
ores' 'Finger' 'Hand'}))

%------------------------------------------------------------------------
% Finger Mean and Variance
Fin_Gen_mean = mean(Gen_Finger_Array);
```

```
Fin_Gen_vari = var(Gen_Finger_Array);
Fin_Imp_mean = mean(Imp_Finger_Array);
Fin_Imp_vari = var(Imp_Finger_Array);


fprintf('\nFinger Mean and Variance:\n')
disp(table({'Genuine';'Imposter'},[Fin_Gen_mean;Fin_Imp_mean],[Fin_Gen_vari;Fin_Imp_vari]
,'VariableNames',{'Finger' 'Mean' 'Variance'}))

%-------------------------------------------------------------------------% Hand Mean
and Variance
Han_Gen_mean = mean(Gen_Hand_Array);
Han_Gen_vari = var(Gen_Hand_Array);
Han_Imp_mean = mean(Imp_Hand_Array);
Han_Imp_vari = var(Imp_Hand_Array);

fprintf('\nHand Mean and Variance:\n')
disp(table({'Genuine';'Imposter'},[Han_Gen_mean;Han_Imp_mean],[Han_Gen_vari;Han_Imp_vari]
,'VariableNames',{'Hand' 'Mean' 'Variance'}))

%-------------------------------------------------------------------------
%D Prime Function

Finger_dprime = DPrime(Fin_Gen_mean,Fin_Imp_mean,Fin_Gen_vari,Fin_Imp_vari);
Hand_dprime = DPrime(Han_Gen_mean,Han_Imp_mean,Han_Gen_vari,Han_Imp_vari);

fprintf('\n D-prime Matcher scores for Hand and Finger:\n')
disp(table({'Finger';'Hand'},[Finger_dprime;Hand_dprime],'VariableNames',{'Scores'
'DPrime'}))
%-------------------------------------------------------------------------
% Histogram Finger
histogram(Gen_Finger_Array);
hold on
histogram(Imp_Finger_Array);


legend('Genuine', 'Imposter')

title('Histogram of Fingers')
xlabel('Number of scores')
ylabel('Scores Distribution')

%Histogram Hand
figure;
histogram(Gen_Hand_Array);
hold on
histogram(Imp_Hand_Array);

legend('Genuine', 'Imposter')

title('Histogram of Hand')
xlabel('Number of scores')
ylabel('Scores Distribution')
%-------------------------------------------------------------------------
% FNMR and FMR

Finger_FNMR = FNMR(Gen_Finger_Array,32);
Finger_FMR = FMR(Imp_Finger_Array,32);

Hand_FNMR = FNMRDIS(Gen_Hand_Array,45);
```

```
Hand_FMR = FMRDIS(Imp_Hand_Array,45);

fprintf('\nFinger and Hand FMR,FNMR Calculations:\n')
disp(table({'Finger';'Hand'},[Finger_FMR;Hand_FMR],[Finger_FNMR;Hand_FNMR],'VariableNames
',{'Threshold' 'FMR' 'FNMR'}))

%-------------------------------------------------------------------------------
%ROC Curves
fprintf('\nCalclating the ROC curve with the given genuine and impostor scores: \n')

in = input('Would you like to provide Genuine and Imposter scores set [Y/N]: ','s');

if(in == 'Y')

    gen = input('Genuine Scores file: ','s');
    imp = input('Imposter Scores file: ','s');

    option = input('What kind of scores are they: \n1)Distance\n2)Similarity\n Type 1 or
2: ','s')

    GEN = fscanf(fopen(gen,'r'),formatSpec);
    IMP = fscanf(fopen(imp,'r'),formatSpec);


    Start_Array = min([GEN;IMP]);
    End_Array = max([GEN;IMP]);

    if(Start_Array == 0)
    Start_Array = 1;
    end
    if(End_Array == 0)
        End_Arrya = 1;
    end

    Hand_Gen_ROC = 1:End_Array;
    Hand_Imp_ROC = 1:End_Array;

    if(option == '1')

        [Hand_Imp_ROC, Han_Gen_ROC] = ROCDIS(GEN,IMP,Start_Array,End_Array);
    elseif(option == '2')
        [Hand_Imp_ROC, Han_Gen_ROC] = ROC(GEN,IMP,Start_Array,End_Array);
    end

    if(EER_Number ~= -1)
        fprintf('EER Number = %d\n',EER_Number);
    else
        fprintf('ERR Number does not exisit!\n');
    end


    plot(Hand_Imp_ROC,Hand_Gen_ROC)

    title('Input ROC Curve')
    xlabel('FMR(FAR)')
    ylabel('FNRM(FRR)')
end

in2 = input('Display the ROC Curves of Hand and Finger? [Y/N]: ','s');
```

```
if(in2 == 'Y')
    fprintf('\nHand and Fingers ROC Curves. \nHand:\n')
    Start_Array = min([Gen_Hand_Array;Imp_Hand_Array]);
    End_Array = max([Gen_Hand_Array;Imp_Hand_Array]);

    if(Start_Array == 0)
        Start_Array = 1;
    end
    if(End_Array == 0)
        End_Arrya = 1;
    end

    Hand_Gen_ROC = 1:End_Array;
    Hand_Imp_ROC = 1:End_Array;

    [Hand_Imp_ROC, Hand_Gen_ROC] =
ROCDIS(Gen_Hand_Array,Imp_Hand_Array,Start_Array,End_Array);

    figure;
    plot(Hand_Imp_ROC,Hand_Gen_ROC)

    title('Hand ROC Curve')
    xlabel('FMR(FAR)')
    ylabel('FNRM(FRR)')


    fprintf('\nFinger:\n')
    Start_Array = min([Gen_Finger_Array;Imp_Finger_Array]);
    End_Array = max([Gen_Finger_Array;Imp_Finger_Array]);

    if(Start_Array == 0)
        Start_Array = 1;
    end
    if(End_Array == 0)
        End_Arrya = 1;
    end

    Fin_Gen_ROC = 1:End_Array;
    Fin_Imp_ROC = 1:End_Array;

    [Fin_Imp_ROC, Fin_Gen_ROC] =
ROC(Gen_Finger_Array,Imp_Finger_Array,Start_Array,End_Array);

    figure;
    plot(Fin_Imp_ROC,Fin_Gen_ROC)

    title('Finger ROC Curve')
    xlabel('FMR(FAR)')
    ylabel('FNRM(FRR)')

    figure;
    x = 0:1;
    y = x;


    plot(Fin_Imp_ROC,Fin_Gen_ROC)
    hold on
    plot(Hand_Imp_ROC,Hand_Gen_ROC)
    hold on
    plot(x,y)
```

```
    grid on


    title('Both ROC Curve')
    legend('Finger', 'Hand')
    xlabel('FMR(FAR)')
    ylabel('FNRM(FRR)')

 end
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
function [ dprime ] = DPrime( mean_gen, mean_imp, var_gen, var_imp )
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here

num = sqrt(2)*abs((mean_gen-mean_imp));
den = sqrt((var_gen)+(var_imp));

dprime = num/den;

end
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
function [ fnmr ] = FNMR( gen_scores ,threshold  )
%UNTITLED3 Summary of this function goes here
%   Detailed explanation goes here
count = 0;
for i = 1:length(gen_scores)
    if(gen_scores(i) < threshold)
        count = count + 1;
    end
end

fnmr = (count/length(gen_scores));
end
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
function [ fmr ] = FMR( imp_scores ,threshold  )
%UNTITLED3 Summary of this function goes here
%   Detailed explanation goes here
count = 0;
for i = 1:length(imp_scores)
    if(imp_scores(i) >= threshold)
        count = count + 1;
    end
end

fmr = (count/length(imp_scores));
end
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
function [fmr_array, fnrm_array] = ROC( scores_gen,scores_imp,start,finish)
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
fnrm_array = finish;
fmr_array = finish;
eer_number = -1;
for i = start:finish
    fmr_array(i) = FMR(scores_imp,i);
    fnrm_array(i) = FNMR(scores_gen,i);
```

```
end

end
%-------------------------------------------------------------------------
%-------------------------------------------------------------------------
function [ fnmr ] = FNMRDIS( gen_scores ,threshold  )
%UNTITLED3 Summary of this function goes here
%   Detailed explanation goes here
count = 0;
for i = 1:length(gen_scores)
    if(gen_scores(i) >= threshold)
        count = count + 1;
    end
end

fnmr = (count/length(gen_scores));
end
%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

function [ fmr ] = FMRDIS( imp_scores ,threshold  )
%UNTITLED3 Summary of this function goes here
%   Detailed explanation goes here
count = 0;
for i = 1:length(imp_scores)
    if(imp_scores(i) < threshold)
        count = count + 1;
    end
end

fmr = (count/length(imp_scores));
end

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------
function [fmr_array, fnrm_array] = ROCDIS( scores_gen,scores_imp,start,finish)
%UNTITLED Summary of this function goes here
%   Detailed explanation goes here
fnrm_array = finish;
fmr_array = finish;
eer_number = -1;
for i = start:finish
    fmr_array(i) = FMRDIS(scores_imp,i);
    fnrm_array(i) = FNMRDIS(scores_gen,i);

end

end
%-------------------------------------------------------------------------
%-------------------------------------------------------------------------
```