ECE 271A Statistical Learning HW3

A53268155 Lien-Hsi Lu

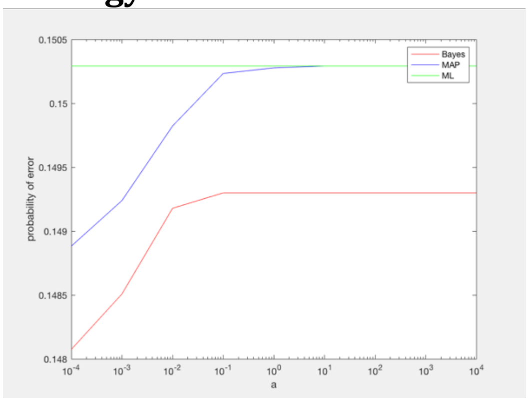
What to hand in: To minimize the number of pages on your report I would advise on handing in, for each dataset and each strategy, a single plot with the curves of classification error as a function of α for: 1) solution based on the predictive equation, 2) MAP solution, and 3) ML solution. The latter will obviously be a constant (horizontal) line. Try to explain 1) the relative behavior of these three curves, 2) how that behavior changes from dataset to dataset, and 3) how all of the above change when strategy 1 is replaced by strategy 2.

Introduction

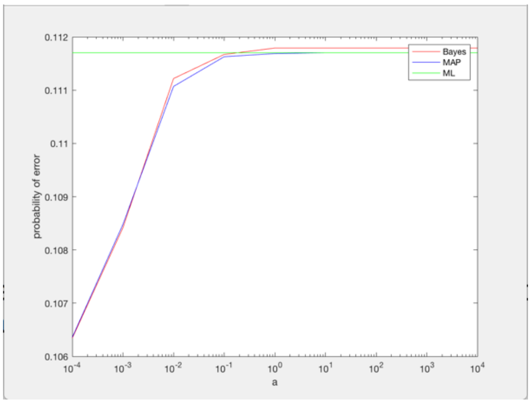
In this homework, we keep on classifying a picture of cheetah and grass. But this time, we use three different solutions (Predictive Equation, MAP solution and ML solution) with four different datasets.

Results:

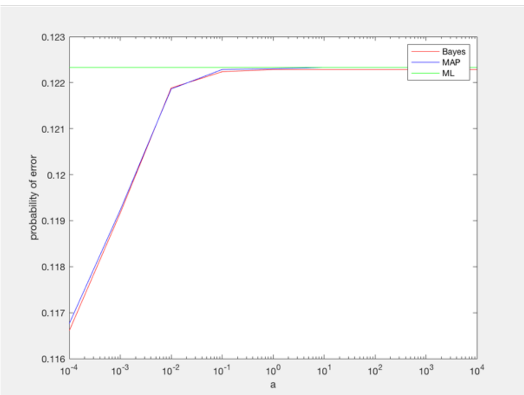
Strategy 1



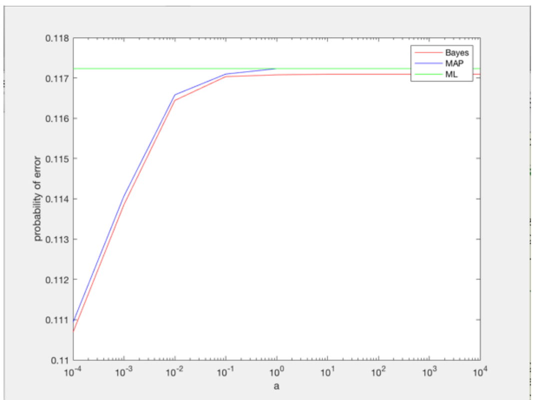
D1



D2

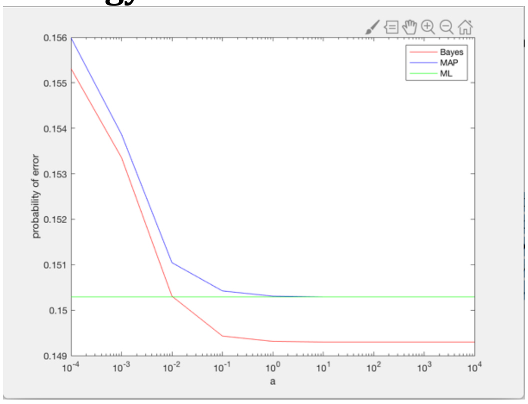


D3

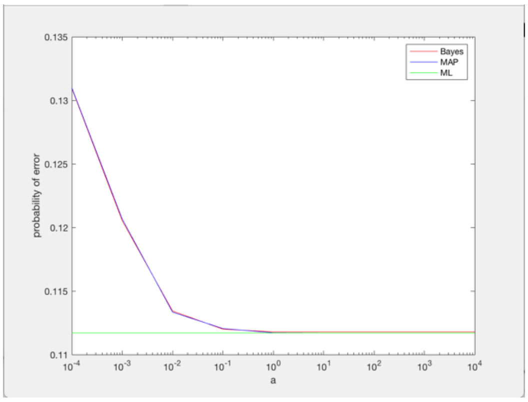


D4

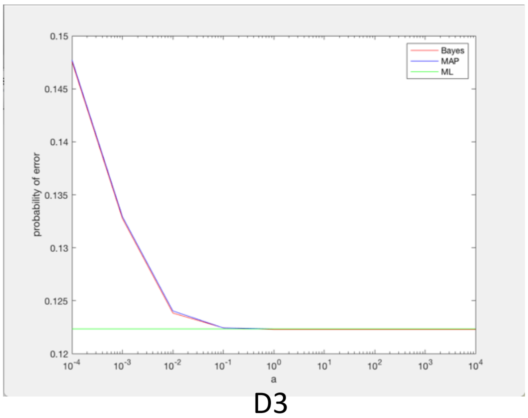
Strategy 2



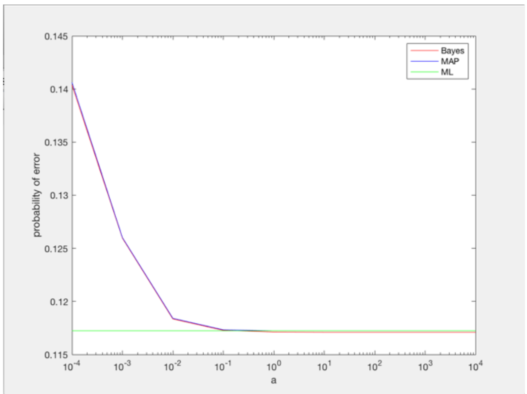
D1



D2



D3



D4

Results Explanation:

1) the relative behavior of these three curves:
   In my observation of four datasets in strategy 1, the curve of Bayesian keep increasing from the lowest probability of error to the value of probability of error of ML. Same as Bayesian, the curve of MAP solution also keep increasing along with increasing α until the value equals to the probability of error to the value of probability of error of ML. It is easy to see that the curve of ML is a horizontal straight line, which is always a constant.
   The predictive distribution: $P_{X|T}(x|D) = G(x, \mu_1, \Sigma + \Sigma_1)$
   The MAP estimate: $P_{X|Y,T}\left(x\middle|D, \theta_i^{MAP}\right) = G(x, \mu_1, \Sigma)$
   The ML solution: $P_{X|Y,T}\left(x\middle|D, \theta_i^{ML}\right) = G(x, \mu, \Sigma)$
   μ_1 is derive from $\mu_1 = \Sigma_0(\Sigma_0 + \frac{1}{n}\Sigma)^{-1}\widehat{\mu_1} + \frac{1}{n}\,\Sigma(\Sigma_0 + \frac{1}{n}\Sigma)^{-1}\mu_0$
   Because of Σ_0 is given as $\Sigma_0 = (\Sigma_0)_{ii} = \alpha w_i$, we can make the upper equation into
$$\mu_1 = \alpha\widehat{\mu_1} + (1-\alpha)\mu_0$$
   As a result, we know that while α becomes bigger, the data will dominate the decision function which make both $P_{X|T}(x|D)$ and $P_{X|Y,T}\left(x\middle|D, \theta_i^{MAP}\right)$ looks quite same with $P_{X|Y,T}\left(x\middle|D, \theta_i^{ML}\right) = G(x, \mu, \Sigma)$. So the both curve will finally go to the ML probability of error. Conversely, small α will make the

prior dominates the decision function. In strategy 1, the prior is 1 for FG and 3 for BG, there proportion is quite similar to our examples of cheetah and grass. In this case, when this prior dominates the decision function, the probability of error will be low because of good separated prior (real world condition). Same as the curves performance in the graphs.

2) how that behavior changes from dataset to dataset:
D1 to D4 are four different datasets and its size gradually gets larger. By seeing the result, the curve of Bayes and MAP are quite the same in D2, D3, D4. The curve of ML has a slight different because it depends on its $G(x, \mu, \Sigma)$, which means sample mean and variance. While α and dataset size gets larger, the three curves has barely difference.
As the equation to derive Σ_1,
$$\Sigma_1 = \Sigma_0(\Sigma_0 + \frac{1}{n}\Sigma)^{-1}\frac{1}{n}\Sigma$$
By seeing this equation, while n increases, Σ_1 will become small which become similar to Σ.
This means $P_{X|T}(x|D) = G(x, \mu_1, \Sigma + \Sigma_1)$ will be similar to $P_{X|Y,T}\left(x\middle|D, \theta_i^{MAP}\right) = G(x, \mu_1, \Sigma)$.
As the equation to derive μ_1,
$$\mu_1 = \Sigma_0(\Sigma_0 + \frac{1}{n}\Sigma)^{-1}\widehat{\mu_1} + \frac{1}{n}\Sigma(\Sigma_0 + \frac{1}{n}\Sigma)^{-1}\mu_0$$
By seeing this equation, while n increases, μ_1 will be similar to $\widehat{\mu_1}$. To sum up, this means when n increases, three curves will be very similar when n and α increases.

3) how all of the above change when strategy 1 is replaced by strategy 2:
In my observation of strategy 2, the curves of Bayesian and MAP keep decreasing while α is getting larger. At last these two curves will be the same value of probability of error of ML curve. In addition, Bayesian has low probability of error while α grows in D1. As mentioned in 1), when α goes big, the data will dominate the decision function. On contrary, the prior will dominate the decision function when α goes small. However, different from strategy 1's prior, strategy 2's prior is 2 for FG and 2 for BG, which is not well-separated, in other words, a far difference between real life. Thus, when α is small, the probability of error is really big. As α grows, the data starts to dominate the decision function, error decreased. The reason of Bayesian has great result in D1 when α grows, because when n (dataset size) is small, $\Sigma + \Sigma_1$ is very different from Σ. So in the graphs after D1, the three curves reach the same value while α gets big.

```
clear all;
load('TrainingSamplesDCT_subsets_8.mat');
alpha = load('Alpha.mat');
prior_1 = load('Prior_1.mat');
prior_2 = load('Prior_2.mat');

%%
Data_BG = D4_BG;
Data_FG = D4_FG;
[m,n] = size(Data_BG);
[p,s] = size(Data_FG);
[a,b] = size(alpha.alpha);
mu_BG = mean(Data_BG);
mu_FG = mean(Data_FG);
p_BG = size(Data_BG,1)/(size(Data_BG,1)+size(Data_FG,1));
p_FG = 1-p_BG;

%Compute covariance BG & FG
cov_BG = zeros(n,n);
cov_FG = zeros(n,n);
for i = 1:m
    cov_BG = cov_BG + (Data_BG(i,:) - mu_BG).' * (Data_BG(i,:) -
mu_BG);
end
    cov_BG = cov_BG/m;
for i = 1:p
    cov_FG = cov_FG + (Data_FG(i,:) - mu_FG).' * (Data_FG(i,:) -
mu_FG);
end
    cov_FG = cov_FG/p;

cov_0 = zeros(n,n);
for k = 1:b %9
    %Compute covariance0
    for h = 1:n %64
        cov_0(h,h) = alpha.alpha(k) * prior_2.W0(h);
    end
```

```
    %Compute mu1,covariance1 of BG & FG
    mu_1_BG =
((cov_0/(cov_0+(1/m)*cov_BG))*mu_BG.'+(1/m)*(cov_BG/(cov_0+(1/m)*cov_
BG))*prior_2.mu0_BG.');
    cov_1_BG = (cov_0/(cov_0+(1/m)*cov_BG))*(1/m)*cov_BG;
    mu_1_FG =
((cov_0/(cov_0+(1/p)*cov_FG))*mu_FG.'+(1/p)*(cov_FG/(cov_0+(1/p)*cov_
FG))*prior_2.mu0_FG.');
    cov_1_FG = (cov_0/(cov_0+(1/p)*cov_FG))*(1/p)*cov_FG;

%%
%input bmp file & ZigZag pattern
A = im2double(imread('cheetah.bmp'));
Z = load('Zig-Zag Pattern.txt');
B = padarray(A,[7,7],'symmetric','post');% original size 255*270 --->
need to use padarray() to fill 255+7, 270+7
[q,l] = size(B);
matrix_zigzag=[];
BAYES = zeros(q,l);
MAP = zeros(q,l);
ML = zeros(q,l);
for i=1:q-7
    for j=1:l-7
        matrix_dct2 = dct2(B(i:i+7,j:j+7));
        matrix_zigzag(Z+ones(8,8)) = matrix_dct2; %store matrix_dct
into zigzag which is from 0~63 so plus a ones(8,8)

%% plug in three solutions to mvnpdf ---> BDR
        cov_sum_BG = cov_1_BG + cov_BG;
        cov_sum_FG = cov_1_FG + cov_FG;
        Pxy_xgrass_BAYES= mvnpdf((matrix_zigzag.'), mu_1_BG,
cov_sum_BG);
        Pxy_xcheeta_BAYES= mvnpdf((matrix_zigzag.'), mu_1_FG,
cov_sum_FG);
        if  (Pxy_xcheeta_BAYES*p_FG) >  (Pxy_xgrass_BAYES*p_BG)
           BAYES(i,j) = 1;
        else% Pxy_xcheeta*p_foreground < Pxy_xgrass*p_background
           BAYES(i,j) = 0;
```

```
        end


        Pxy_xgrass_MAP= mvnpdf((matrix_zigzag.'), mu_1_BG, cov_BG);
        Pxy_xcheeta_MAP= mvnpdf((matrix_zigzag.'), mu_1_FG, cov_FG);
        if  (Pxy_xcheeta_MAP*p_FG) >  (Pxy_xgrass_MAP*p_BG)
          MAP(i,j) = 1;
        else% Pxy_xcheeta*p_foreground < Pxy_xgrass*p_background
          MAP(i,j) = 0;
        end


        Pxy_xgrass_ML= mvnpdf(matrix_zigzag, mu_BG, cov_BG);
        Pxy_xcheeta_ML= mvnpdf(matrix_zigzag, mu_FG, cov_FG);
        if  (Pxy_xcheeta_ML*p_FG) >  (Pxy_xgrass_ML*p_BG)
            ML(i,j) = 1;
        else% Pxy_xcheeta*p_foreground < Pxy_xgrass*p_background
            ML(i,j) = 0;
        end
    end
end
%% Error Rate of BAYES
groundtruth_mask = im2double(imread('cheetah_mask.bmp'));
diff_grass = 0;
diff_cheeta = 0;
count_grass = 0;
count_cheeta = 0;
for i=1:(q-7)
    for j=1:(l-7)
        if BAYES(i,j) == 1
            if groundtruth_mask(i,j) == 0
                diff_cheeta = diff_cheeta + 1;
            end
        elseif BAYES(i,j) == 0
            if groundtruth_mask(i,j) == 1
                diff_grass = diff_grass + 1;
            end
        end
```

```
     end
end

for i=1:(q-7)
    for j=1:(l-7)
        if groundtruth_mask(i,j) == 1
            count_cheeta = count_cheeta + 1;
        else
            count_grass = count_grass + 1;
        end
    end
end
error_cheeta = (diff_cheeta / count_cheeta)*p_FG ;
error_grass = (diff_grass / count_grass)*p_BG;
error_BAYES(k) = error_cheeta + error_grass;

%% Error Rate of MAP
%groundtruth_mask = im2double(imread('cheetah_mask.bmp'));
diff_grass = 0;
diff_cheeta = 0;
count_grass = 0;
count_cheeta = 0;
for i=1:(q-7)
    for j=1:(l-7)
        if MAP(i,j) == 1
            if groundtruth_mask(i,j) == 0
                diff_cheeta = diff_cheeta + 1;
            end
        elseif MAP(i,j) == 0
            if groundtruth_mask(i,j) == 1
                diff_grass = diff_grass + 1;
            end
        end
     end
end

for i=1:(q-7)
    for j=1:(l-7)
```

```
        if groundtruth_mask(i,j) == 1
            count_cheeta = count_cheeta + 1;
        else
            count_grass = count_grass + 1;
        end
    end
end
error_cheeta = (diff_cheeta / count_cheeta)*p_FG ;
error_grass = (diff_grass / count_grass)*p_BG;
error_MAP(k) = error_cheeta + error_grass;

%% Error Rate of ML
%groundtruth_mask = im2double(imread('cheetah_mask.bmp'));
diff_grass = 0;
diff_cheeta = 0;
count_grass = 0;
count_cheeta = 0;
for i=1:(q-7)
    for j=1:(l-7)
        if ML(i,j) == 1
            if groundtruth_mask(i,j) == 0
                diff_cheeta = diff_cheeta + 1;
            end
        elseif ML(i,j) == 0
            if groundtruth_mask(i,j) == 1
                diff_grass = diff_grass + 1;
            end
        end
     end
end

for i=1:(q-7)
    for j=1:(l-7)
        if groundtruth_mask(i,j) == 1
            count_cheeta = count_cheeta + 1;
        else
            count_grass = count_grass + 1;
        end
```

```
    end
end
error_cheeta = (diff_cheeta / count_cheeta)*p_FG ;
error_grass = (diff_grass / count_grass)*p_BG;
error_ML(k) = error_cheeta + error_grass;

end



x = alpha.alpha;
yB = error_BAYES;
plot(x,yB,'r');
set(gca, 'XScale', 'log');
hold on;

yMAP = error_MAP;
plot(x,yMAP,'b');
set(gca, 'XScale', 'log');
hold on;

yML = error_ML;
plot(x,yML,'g');
set(gca, 'XScale', 'log');
hold off;
xlabel('a');
ylabel('probability of error');
legend('Bayes','MAP','ML');
```