ECE 271A- HW3/HW4

Reya Sadhu
A59026753

Given,
$P_{x|\mu,\Sigma} = G(x, \mu, \Sigma), where\ \Sigma =$ sample covariance

$P_{\mu}(\mu) = G(\mu, \mu_0, \Sigma_0), where\ \mu_0\ is\ given, (\Sigma_0)_{ii} = \alpha w_i, with\ given\ \alpha\ and\ w$

So, the posterior distribution is,

$$P_{\mu|T}(\mu, D) \propto \prod_i P_{x|\mu}(D|\mu)\, P_{\mu}(\mu)$$

$$\propto \prod_i G(x_i, \mu, \Sigma)\, G(\mu, \mu_0, \Sigma_0)$$

$$\propto \exp\,(-\frac{1}{2}\left[(\mu - \mu_0)^T {\Sigma_0}^{-1}(\mu - \mu_0) + \sum_{i=1}^{N}(x_i - \mu)^T \Sigma^{-1}(x_i - \mu)\right])$$

Ignoring all the terms without μ,

$$\propto \exp\left(-\frac{1}{2}\left[\mu^T {\Sigma_0}^{-1}\mu - 2\mu^T {\Sigma_0}^{-1}\mu_0 - 2\mu^T \Sigma^{-1}\sum_{i=1}^{N} x_i + N\mu^T \Sigma^{-1}\mu\right]\right)$$

$$= \exp\,(-\frac{1}{2}[\mu^T({\Sigma_0}^{-1} + N\Sigma^{-1})\mu - 2\mu^T\left({\Sigma_0}^{-1}\mu_0 + \Sigma^{-1}\sum_{i=1}^{N} x_i\right)])$$

Multiplying a constant term to express this as a normal equation, it can be written as,

$$\propto \exp\,(-\frac{1}{2}(\mu - \mu_1)^{\,T} {\Sigma_1}^{-1}(\mu - \mu_1)$$

Where,

$$\Sigma_1 = ({\Sigma_0}^{-1} + N\Sigma^{-1})^{-1}$$

$$\mu_1 = ({\Sigma_0}^{-1} + N\Sigma^{-1})^{-1}(\left({\Sigma_0}^{-1}\mu_0 + \Sigma^{-1}\sum_{i=1}^{N} x_i\right)$$

Using the matrix identity,

$$(A^{-1} + B^{-1})^{-1} = A(A + B)^{-1}B = B(A + B)^{-1}A$$

These two parameters can be written as,

$$\Sigma_1 = \Sigma_0(\Sigma_0 + \frac{1}{N}\Sigma)^{-1}\frac{1}{N}\Sigma$$
$$\mu_1 = \Sigma_0(\Sigma_0 + \frac{1}{N}\Sigma)^{-1}\left(\frac{1}{N}\sum_{i=1}^{N} x_i\right) + \frac{1}{N}\Sigma(\Sigma_0 + \frac{1}{N}\Sigma)^{-1}\mu_0$$

So, the posterior distribution $P_{\mu|T}(\mu, D) \sim G(\mu, \mu_1, \Sigma_1)$

Now, the predictive distribution is,

$$P_{x|T}(x|D) = \int P_{x|\mu}(x|\mu)P_{\mu|T}(\mu|D)\, d\mu$$
$$= \int f(x-\mu)h(\mu)d\mu$$
$$With, f(x) = G(x, 0, \Sigma) and\ h(x) = G(x, \mu_1, \Sigma_1)$$

So, the posterior is the just the sum of the two independent Normal variables, so it also follows a normal distribution with mean equal to the sum of the two means and covariance as the sum of the two variables.

So, $P_{x|T}(x|D) \sim G(x, \mu_1, \Sigma + \Sigma_1)$

Plugging this into the Bayesian Decision Rule,

$$i^*(x) = \underset{i}{arg\ max}\ P_{i|X}(i|x) \ = \ \underset{i}{arg\ max}\ P_{X|i}(x|i)P(i)$$

$$P_{X|i}(x|i, D) = G(x, \mu_{1_i}, \Sigma_i + \Sigma_{1_i})$$

For the Maximum likelihood estimator,

$$\mu_{ML} = Sample\ mean = \frac{1}{N}\sum_{i=1}^{N} x_i$$

And for this case, the BDR becomes,

$$i^*(x) = \underset{i}{arg\ max}\ P_{X|i}(x|i;\mu_{ML})P(i)$$

$$P_{X|i}(x|i;\mu_{ML}) = G(x, \mu_{ML_i}, \Sigma_i)$$

For the MAP estimate,

$$\mu_{MAP} = \underset{\mu}{arg\ max}\ P_{\mu|T}(\mu|D), which\ follows\ G(\mu, \mu_1, \Sigma_1)$$

As it's a Gaussian distribution, mode=mean=μ_1

So, $\mu_{MAP} = \mu_1$

So, the BDR is,

$$i^*(x) = \underset{i}{arg\ max}\ P_{X|i}(x|i;\mu_{MAP})P(i)$$

$$P_{X|i}(x|i;\mu_{MAP}) = G(x, \mu_{1_i}, \Sigma_i)$$

This BDR can be expressed as,

$$g_{FG}(x) = \frac{1}{1 + exp(d_{FG}(x - \mu_{FG}) - d_{BG}(x - \mu_{BG}) + \alpha_{FG} - \alpha_{BG})}$$

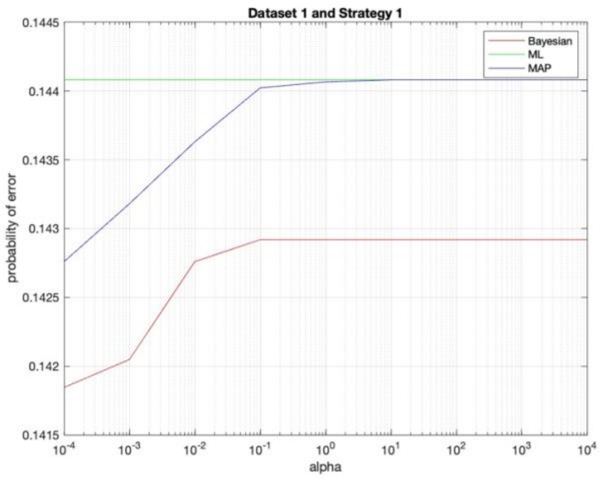
$$where, \qquad d_i(x, y) = (x - y)^T \Sigma_i^{-1}(x - y)$$
$$\alpha_i = log(2\pi)^d |\Sigma_i| - 2logP(i)$$

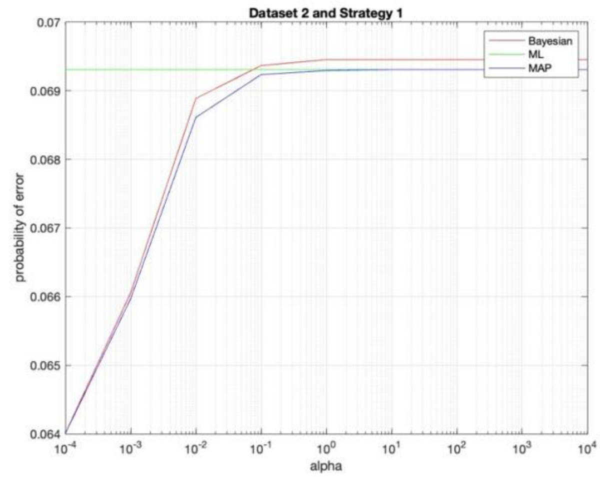
If $g_{FG}(x) > 0.5, classify\ the\ pixel\ as\ Foreground/cheetah.$

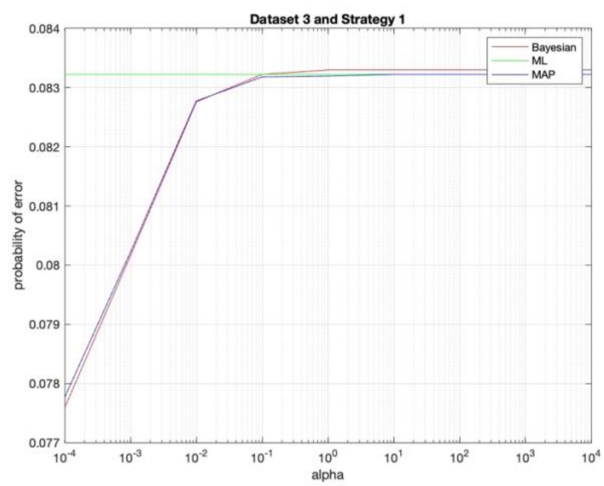
The four datasets used in the analysis,

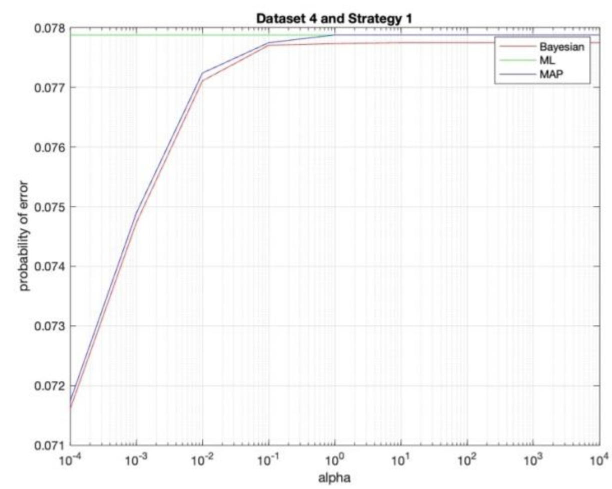
Dataset	Cheetah example	Grass example
D_1	75	300
D_2	125	500
D_3	175	700
D_4	225	900

Figure 1 to 4 show the probability of error as a function of α, based on the dataset D1 and the first strategy where $\mu_{0_{FG}} = 1$ and $\mu_{0_{BG}} = 3$ on these three different methods, Bayesian, ML and MAP.









The ML solution is derived only from the data and has no association with the prior and thus α, and so its constant for all values of α.

As a general trend, the Bayesian and MAP converges to ML as α increases. This is because as the prior covariance is directly proportional to α, it becomes larger and thus the prior becomes less reliable as α increases.

The Bayesian solution use prior information as well as info from the data and the MAP solution uses the prior information to some extent. Here the priors have different means, meaning they provide valuable information of the model, and relying less on the prior results in worse performance. Thus, the MAP and Bayesian error increase with α.

And when the prior covariance is much larger than the sample covariance, the mean converges to the ML estimate for both Bayesian and MAP. The Bayesian covariance also

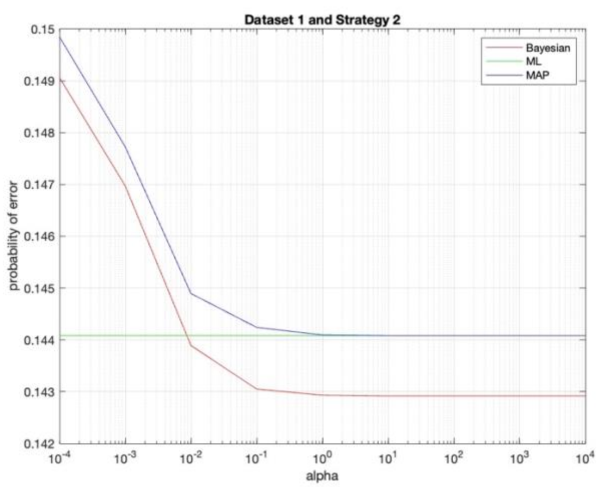
converges to the ML estimate (sample covariance). Thus, both of them converge to the ML model.
On the other hand, the prior helps a lot when α is small and that's why for small α, the Bayesian and MAP error is much lesser than the ML error.

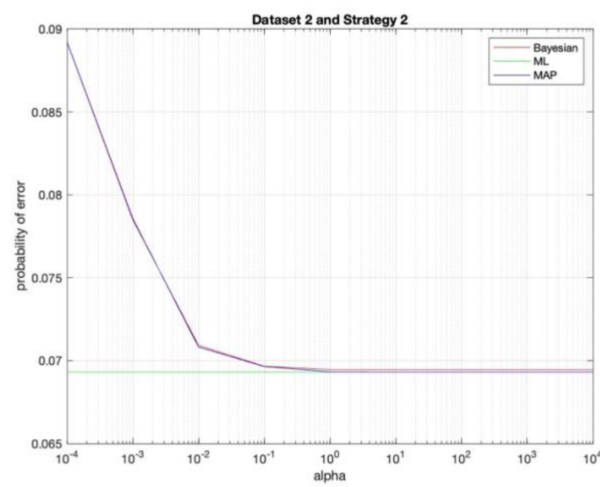
Also, we can see the smallest dataset (D_1) has error which is almost double of the error of the largest dataset (D_4). This shows that as the dataset increases the modelling error decreases.
Precisely, the errors of different datasets are sorted like this,

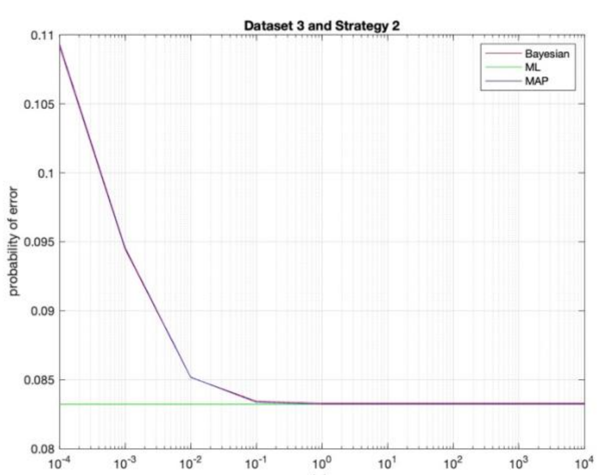
$$D_2 < D_3 < D_4 < D_1$$

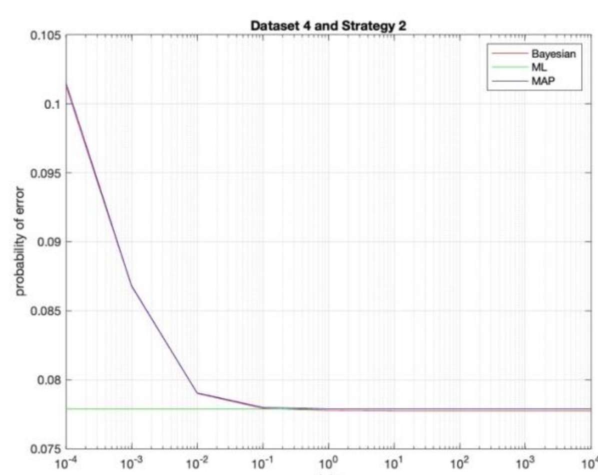
Ideally, the error should decrease with increase in dataset size. But here the datasets are not sufficiently larger than each other, so its plausible to have this order.

The following four graphs are for strategy 2:









For Strategy 2, **the priors are same for both the class**, so the prior give no valuable information of the model.

So, not relying on the prior will give a better classification and so, as prior variance increases with α, the MAP and Bayesian rely less on prior and give a better classification (less probability of error).
Also, as MAP and Bayesian use the prior model, which is not precise, generally they perform worse (larger probability of error) than the ML model.
Similarly, both the model converges to ML with increase in α, as they rely less and less on the prior belief.

Here also, the smallest dataset has larger error than the largest dataset. And the dataset not being sufficiently larger than each other, the error trend does the dataset size precisely.

Code:

```
load('/Users/reyasadhu/Downloads/hw3Data/TrainingSamplesDCT_subsets_8.mat');
load('/Users/reyasadhu/Downloads/hw3Data/Alpha.mat');

train_BG = D1_BG;
train_FG = D1_FG;
helper(1,train_BG,train_FG,alpha,1);
helper(1,train_BG,train_FG,alpha,2);

train_BG = D2_BG;
train_FG = D2_FG;
helper(2,train_BG,train_FG,alpha,1);
helper(2,train_BG,train_FG,alpha,2);

train_BG = D3_BG;
train_FG = D3_FG;
helper(3,train_BG,train_FG,alpha,1);
helper(3,train_BG,train_FG,alpha,2);

train_BG = D4_BG;
train_FG = D4_FG;
```

```
helper(4,train_BG,train_FG,alpha,1);
helper(4,train_BG,train_FG,alpha,2);

function []= helper(Data, BG, FG, alpha, strategy)

    p=load(['/Users/reyasadhu/Downloads/hw3Data/Prior_',num2str(strategy),'.mat']);
    w0=p.W0;
    mu0_FG=p.mu0_FG;
    mu0_BG=p.mu0_BG;

    % ML Priors
    Prior_FG=size(FG,1)/(size(FG,1)+size(BG,1));
    Prior_BG=size(BG,1)/(size(FG,1)+size(BG,1));

    mu_FG=mean(FG,1);
    mu_BG=mean(BG,1);

    FG_cov = cov(FG) * (size(FG, 1) - 1) / size(FG, 1);
    BG_cov = cov(BG) * (size(BG, 1) - 1) / size(BG, 1);

    Error_bayesian = zeros(1,9);
    Error_ML = zeros(1,9);
    Error_MAP = zeros(1,9);

    for k=1:length(alpha)

        cov_0=diag(w0)*alpha(k);

        N_FG=size(FG,1);
        sample_cov_FG=FG_cov/N_FG;
        mu_1_FG=cov_0/(cov_0+sample_cov_FG)*mu_FG'+sample_cov_FG/(cov_0+sample_cov_FG)*mu0_FG';
        cov_1_FG=cov_0/(cov_0+sample_cov_FG)*sample_cov_FG;
```

```
N_BG=size(BG,1);
sample_cov_BG=BG_cov/N_BG;
mu_1_BG=cov_0/(cov_0+sample_cov_BG)*mu_BG'+sample_cov_BG/(cov_0+sample_cov_BG)*mu0_BG';
cov_1_BG=cov_0/(cov_0+sample_cov_BG)*sample_cov_BG;



%Predictive Distribution
mu_pred_FG=mu_1_FG;
mu_pred_BG=mu_1_BG;
cov_pred_FG=cov_1_FG+FG_cov;
cov_pred_BG=cov_1_BG+BG_cov;

% Bayesian

I = imread('/Users/reyasadhu/Downloads/homework1/cheetah.bmp');
I=im2double(I);
A_Bayesian = zeros((size(I, 1) - 7),(size(I, 2) - 7));
A_ML= zeros((size(I, 1) - 7),(size(I, 2) - 7));
A_MAP = zeros((size(I, 1) - 7),(size(I, 2) - 7));

for i = 1:size(I, 1) - 7
   for j = 1:size(I, 2) - 7

        block = I(i:i+7, j:j+7);
        block_dct = dct2(block);
        dct_flat=zigzaged(block_dct);

        %Bayesian
        alp_FG=log(((2*pi)^64)*det(cov_pred_FG))-2*log(Prior_FG);
        alp_BG=log(((2*pi)^64)*det(cov_pred_BG))-2*log(Prior_BG);
        g_cheetah=1/(1+exp(dxy(dct_flat',mu_pred_FG,cov_pred_FG)-
dxy(dct_flat',mu_pred_BG,cov_pred_BG)+alp_FG-alp_BG));
```

```
            if g_cheetah>0.5
               A_Bayesian(i+3,j+3)=1;
            end

            %ML
            alp_FG=log(((2*pi)^64)*det(FG_cov))-2*log(Prior_FG);
            alp_BG=log(((2*pi)^64)*det(BG_cov))-2*log(Prior_BG);
            g_cheetah=1/(1+exp(dxy(dct_flat',mu_FG',FG_cov)-dxy(dct_flat',mu_BG',BG_cov)+alp_FG-alp_BG));
            if g_cheetah>0.5
               A_ML(i+3,j+3)=1;
            end

            %MAP
            alp_FG=log(((2*pi)^64)*det(FG_cov))-2*log(Prior_FG);
            alp_BG=log(((2*pi)^64)*det(BG_cov))-2*log(Prior_BG);
            g_cheetah=1/(1+exp(dxy(dct_flat',mu_pred_FG,FG_cov)-dxy(dct_flat',mu_pred_BG,BG_cov)+alp_FG-
alp_BG));
            if g_cheetah>0.5
               A_MAP(i+3,j+3)=1;
            end

         end
      end

      %Padding
      A_Bayesian_resized=zeros(255,270);
      A_ML_resized=zeros(255,270);
      A_MAP_resized=zeros(255,270);
      for i=4:251
         for j=4:266
            A_Bayesian_resized(i,j)=A_Bayesian(i-3,j-3);
            A_ML_resized(i,j)=A_ML(i-3,j-3);
            A_MAP_resized(i,j)=A_MAP(i-3,j-3);
```

```
            end
        end

        Error_bayesian(k)=error(A_Bayesian_resized);
        Error_ML(k)=error(A_ML_resized);
        Error_MAP(k)=error(A_MAP_resized);




    end

    fig=figure;
    semilogx(alpha, Error_bayesian, '-r',alpha, Error_ML, '-g',alpha, Error_MAP, '-b');
    grid
    %ylim([0.1460 0.1500])
    legend('Bayesian','ML','MAP')
    xlabel('alpha');
    ylabel('probability of error');
    title(['Dataset ',num2str(Data),' and Strategy ',num2str(strategy)]);
    print(fig,'-djpeg',sprintf("/Users/reyasadhu/Desktop/Masters/ECE 271A Statistical Learning/HW3/Dataset %d
strategy %d.jpg",Data,strategy));
end

function output= zigzaged(input)
    zigzag=importdata('/Users/reyasadhu/Downloads/homework1/Zig-Zag Pattern.txt') ;
    zigzag=zigzag+1;
    output=zeros(1,64);
    for i=1:8
        for j=1:8
            output(zigzag(i,j))=input(i,j);
        end
    end
end
```

```
function output=dxy(x,y,cov)
   output=transpose(x-y)/cov*(x-y);
end

function prob_err=error(input)
   im_test = imread('/Users/reyasadhu/Downloads/homework1/cheetah_mask.bmp');
   im_test=im2double(im_test);
   err=abs(im_test-input);
   prob_err=sum(err,"all")/(255*270);
end
```