Final Exam

IsyE 8843

Ying Hung
901984034

Problem 1

There are many application of baysian wavelet in speech data analysis. The testing speech data include one clean speech utterance and three levels of noisy utterance. Here, the clean data is recorded by myself when I was saying ”Two”. Noisy data were made by integrated artificial white noise onto the clean data according to different S/N ratio. For the denoise part, I use filter ”Vaidyanathan”.

When I apply Bayesian Shrinkage to this problem, the likelihood of a detail wavelet coefficient is $N\left(\theta,\sigma^2\right)$, and that the prior on θ is also $N\left(0,\tau^2\right)$. The Bayes rule is: $\tau^2/\left(\tau^2+\sigma^2\right)d$. Take $\tau^2=0.01$ and $\sigma^2=0.1$.

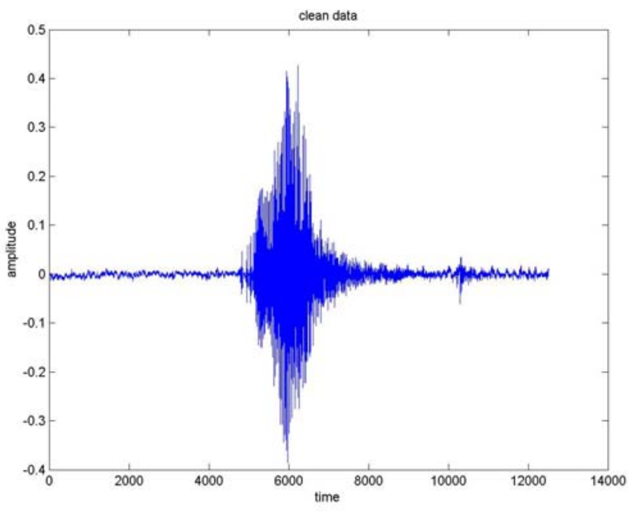


Figure 1 : clean speech data pronouncing “Two”.

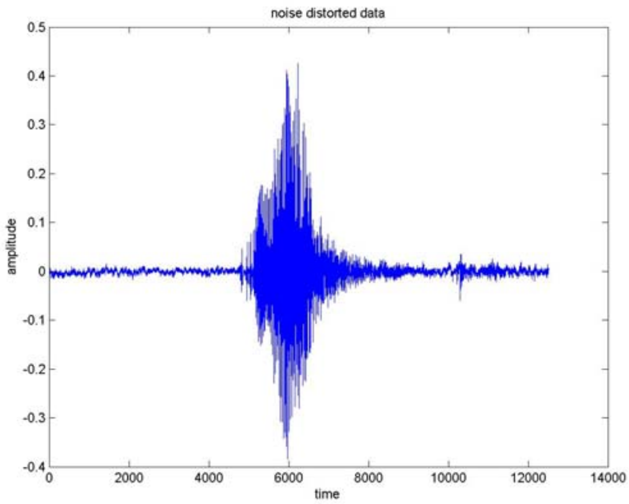


Figure 2: Noisy data by adding white noise (S/N ratio=0.05/0.11)

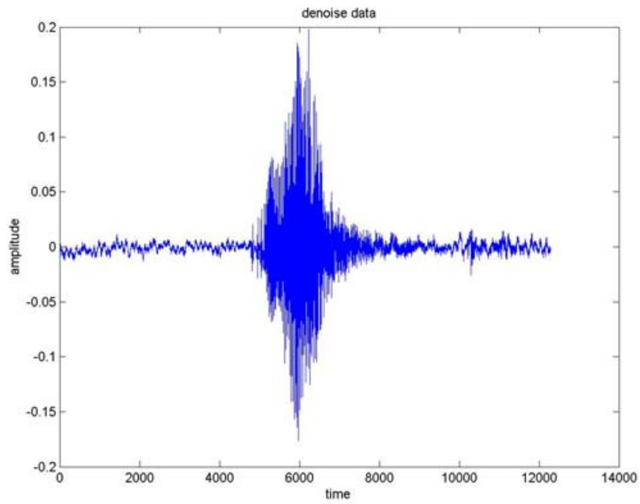
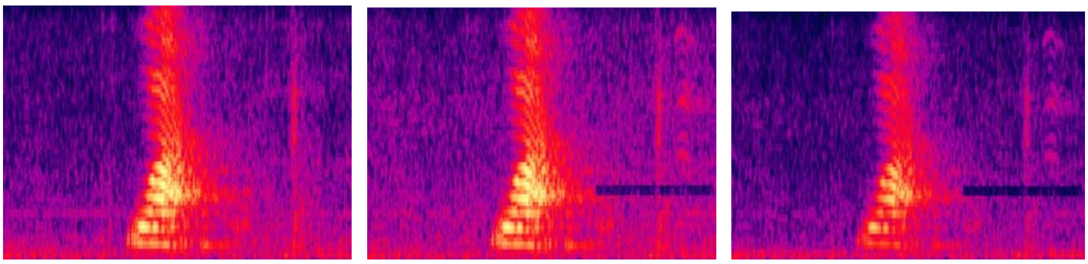


Figure 3: denoised data

We can look at the performance of these three data by looking at the waveform.



(a) clean data (b) noisy data (c) denoised data

Figure 4: waveform performance

We can see that after apply Bayes shinkage, the denoised data is much clear and close to the original clean data.

Problem 2

(1) By exact calculation, we have

$$P(J1,M1|B1)=\frac{1}{P(B1)}\sum_{E,A}P(B1,J1,M1,E,A)$$

$$=\frac{1}{P(B1)}\sum_{E,A}P(A|B1,E)\cdot P(B1)\cdot P(E)\cdot P(M1|A)\cdot P(J1|A)$$

$$=\frac{1}{P(B1)}\cdot\sum_{A}\left[\sum_{E}P(A|B1,E)\cdot P(E)\right]\cdot P(B1)\cdot P(M1|A)\cdot P(J1|A)$$

$$=\sum_{A}\left[\sum_{E}P(A|B1,E)\cdot P(E)\right]\cdot P(M1|A)\cdot P(J1|A)$$

And we can calculate

$$P(A1|B1)=P(A1|B1,E0)P(E0)+P(A1|B1,E1)P(E1)$$

$$=0.998\times 0.94+0.002\times 0.95=0.94002$$

$$P(A0|B1)=P(A0|B1,E0)P(E0)+P(A0|B1,E1)P(E1)$$

$$=0.998\times 0.06+0.002\times 0.05=0.05998$$

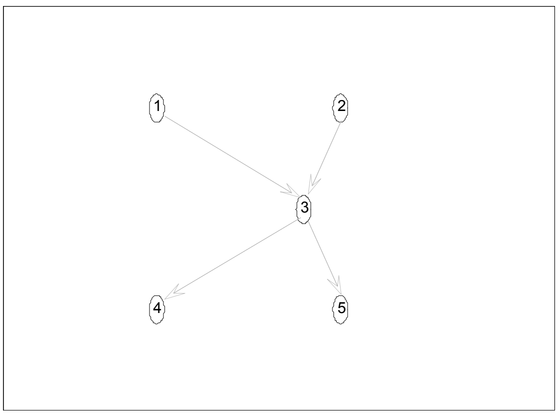
So,

$$P(J1,M1|B1)=0.05998\times 0.01\times 0.05+0.94002\times 0.7\times 0.9$$

$$=0.5922426$$

(2) By using Kevin Murphy's BNT,

The following is plot of the relation in BNT



And the conditional probability is:

$$P(J1|M1,B1) = 0.8992 \qquad P(M1|B1) = 0.6586$$

$$P(J1,M1|B1) = 0.5922$$

(3) Using BUGS we have:

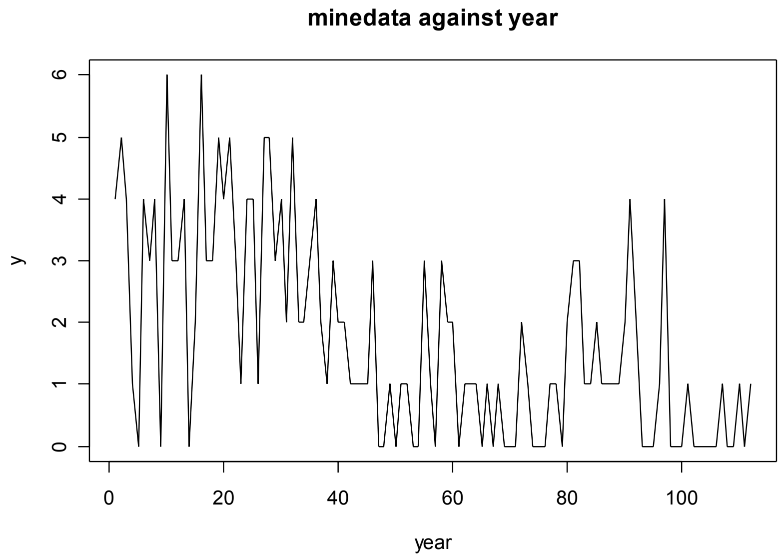
$$P(J1|M1,B1) = 0.898 \qquad P(M1|B1) = 0.655$$

So, $P(J1,M1|B1) = P(J1|M1,B1) \cdot P(M1|B1)$

$= 0.898*0.655 = 0.5882$

Problem 3

From the data during the period 1851 to 1962, we plot the minedata Y against year,



From this plot we can see that there is a change point located around year 1890, it is the 40th year in the period. After assume themodel, we can calculate the full conditionals like the following:

$$f\left(Y,\lambda,\mu,\tau\right)=f\left(Y,\lambda,\mu|\tau\right)\cdot f\left(\tau\right)$$

$$=\left\{\left[\prod_{i=1}^{\tau} f\left(Y_i|\tau,\lambda\right)\right]\cdot f\left(\lambda\right)\cdot\left[\prod_{i=\tau+1}^{n} f\left(Y_i|\tau,\mu\right)\right]\cdot f\left(\mu\right)\right\}\cdot f\left(\tau\right)$$

$$=\left\{\left[\prod_{i=1}^{\tau}\frac{e^{-\lambda}\lambda^{y_i}}{Y_i\,!}\right]\cdot\frac{\beta_\lambda^{\alpha_\lambda}\lambda^{\alpha_\lambda-1}e^{-\lambda\beta_\lambda}}{\Gamma\left(\alpha_\lambda\right)}\cdot\left[\prod_{i=\tau+1}^{n}\frac{e^{-\mu}\mu^{y_i}}{Y_i\,!}\right]\cdot\frac{\beta_\mu^{\alpha_\mu}\mu^{\alpha_\mu-1}e^{-\mu\beta_\mu}}{\Gamma\left(\alpha_\mu\right)}\right\}\cdot\frac{1}{n}$$

$$\infty\ \lambda^{\alpha_\lambda+\sum_{i=1}^{\tau}Y_i-1}\times e^{-(\tau+\beta_\lambda)\lambda}\times\mu^{\alpha_\mu+\sum_{i=\tau+1}^{n}Y_i-1}\times e^{-((n-\tau)+\beta_\mu)\mu}$$

Let, $$g\left(\tau\right)=\lambda^{\sum_{i=1}^{\tau}Y_i}\times\mu^{\sum_{i=\tau+1}^{n}Y_i}\times e^{-(\lambda-\mu)\tau}$$

We have,

$$f\left(\tau|Y,\lambda,\mu\right)=\frac{g\left(\tau\right)}{\sum_{\tau=1}^{n}g\left(\tau\right)}$$

$$f\left(\lambda|\tau,\mu,Y\right)=Gamma\left(\alpha_\lambda+\sum_{i=1}^{\tau}Y_i,\beta_\lambda+\tau\right)$$

$$f(\mu|\tau,\lambda,Y) = Gamma\left(\alpha_\mu + \sum_{i=\tau+1}^{n} Y_i, \beta_\mu + (n-\tau)\right)$$

Because in the first part of data($Y_1 \cdots Y_{40}$), mean and variance are λ , and on the other hand λ comes from Gamma distribution with parameter $\alpha_\lambda, \beta_\lambda$. So, we can use mean and variance of ($Y_1 \cdots Y_{40}$) as two observation of λ . Moreover,

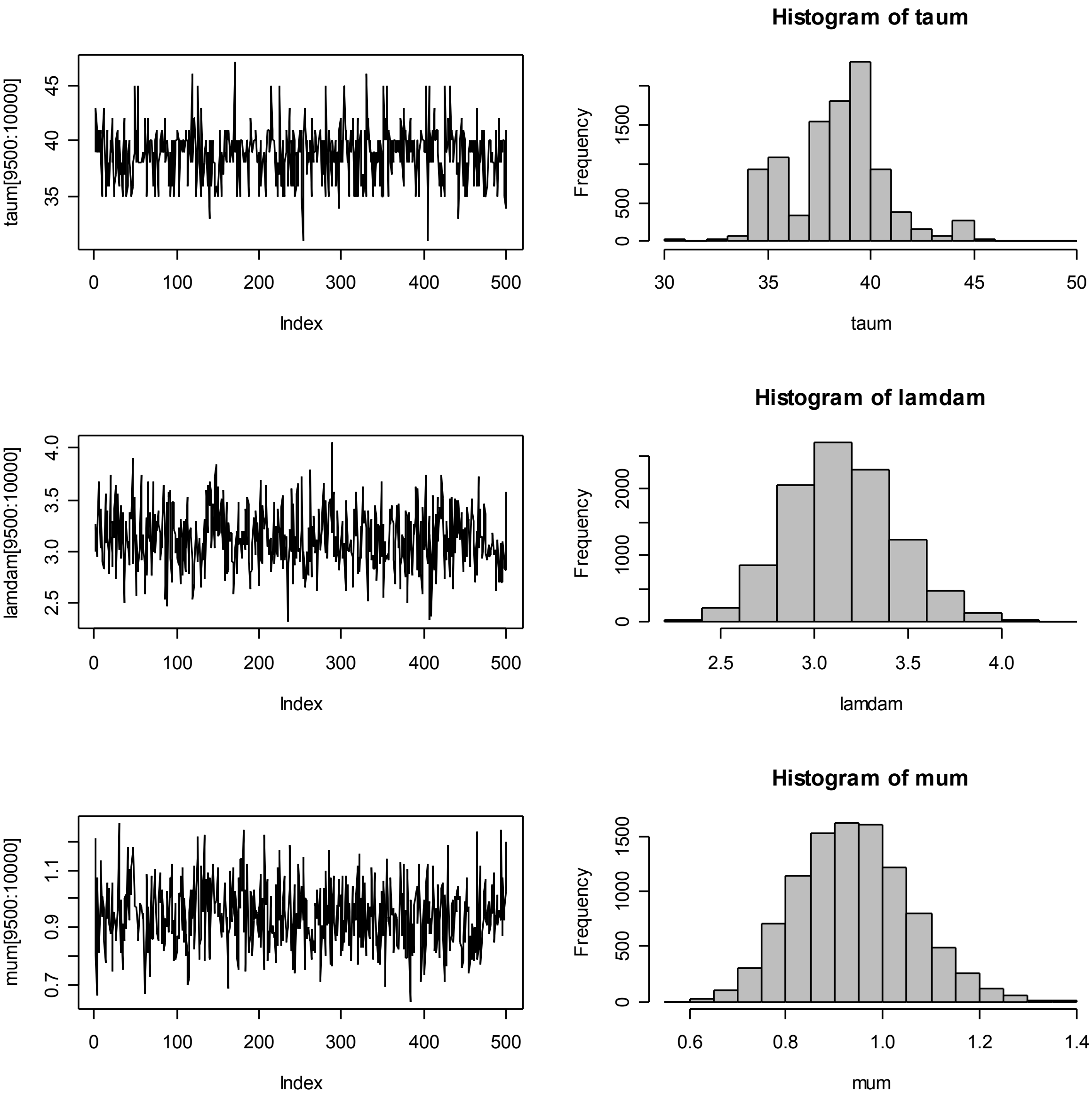
$$E(\lambda) = \alpha_\lambda / \beta_\lambda, Var(\lambda) = \alpha_\lambda / \beta_\lambda^2,$$

we can use moment method to estimate $\alpha_\lambda, \beta_\lambda$.

After 10000 simulation by using Gibbs sampling with the initial value mentioned above, we have the following estimate.

	Mean(Std)
τ	38.8258(2.40)
λ	3.05154(0.24)
μ	0.96847(0.08)

So, the change point is happened around 39. I have try different kind of initial value, there are no significant effect on the estimate of these three parameters. The following is the performance of these parameters in the last 500 simulation.



taum[9500:10000]
35
40
45
0
100
200
300
400
500
Index
Histogram of taum
Frequency
0
500
1500
30
35
40
45
50
taum
lamdam[9500:10000]
2.5
3.0
3.5
4.0
0
100
200
300
400
500
Index
Histogram of lamdam
Frequency
0
1000
2000
2.5
3.0
3.5
4.0
lamdam
mum[9500:10000]
0.7
0.9
1.1
0
100
200
300
400
500
Index
Histogram of mum
Frequency
0
500
1000
1500
0.6
0.8
1.0
1.2
1.4
mum

R-code for Problem 3

```
#########################
y= c( 4,5,4,1,0,4,3,4,0,6,3,3,4...)
##### function to generate tau #####
g=function(lamda, mu,k,y)
{ gv=lamda^(sum(y[1:k]))*mu^(sum(y)-sum(y[1:k]))*exp(-(lamda-mu)*k)
  return(gv)}
tau=function(r,lamda,mu,y)
{    k=1;sum=0;sc=0
        for(i in 1:n)
      {    sum=sum+g(lamda,mu,i,y)}
        for(i in 1:n)
        {
             sc=sc+g(lamda,mu,i,y)
             if (sc<=r*sum)
             k=i
        }
        return(k)
}
plot(y,type="l")
#########################
n=112;
taum=matrix(nrow=10000,rep(0,10000))
lamdam=matrix(nrow=10000,rep(0,10000))
mum=matrix(nrow=10000,rep(0,10000))
taum[1]=40
lamdam[1]=(sum(y[1:taum[1]]))/taum[1]
mum[1]=(sum(y)-sum(y[1:taum[1]]))/(n-taum[1])
#######initial ##########
#    use two parts (1-40,40-112) of data to
#    guess    the initial of beta and alpha
#########################
PI=y[1:40]
PII=y[41:112]
lamda1=mean(PI);lamda2=var(PI) ### sample of lamda
mu1=mean(PII);mu2=var(PII) #### sample of mu
beta.lamda=mean(c(lamda1,lamda2))/var(c(lamda1,lamda2))
beta.mu=mean(c(mu1,mu2))/var(c(mu1,mu2))
```

```
alpha.lamda=mean(c(lamda1,lamda2))^2/var(c(lamda1,lamda2))
alpha.mu=mean(c(mu1,mu2))^2/var(c(mu1,mu2))
####### main part ########
for(i in 2: 10000)
{
  r=runif(1)
  taum[i]=tau(r,lamdam[i-1],mum[i-1],y)
  lamdam[i]=rgamma(1,alpha.lamda+sum(y[1:taum[i-1]]),rate =
(beta.lamda+taum[i-1]))
  mum[i]=rgamma(1,alpha.mu+sum(y)-sum(y[1:taum[i-1]]),rate =
(beta.mu+n-taum[i-1]))
}######## plot ###########
par(mfrow=c(3,2))
plot(taum[9500:10000],type="l");hist(taum,col="gray")
plot(lamdam[9500:10000],type="l");hist(lamdam,col="gray")
plot(mum[9500:10000],type="l");hist(mum,col="gray")
```

Matlab code for problem 2

```
#####################
matlab code
N = 5;
dag = zeros(N,N);
B = 1; E = 2; A = 3; J = 4; M = 5;
dag([B,E],A) = 1;
dag(A,[J,M]) = 1;
node_sizes = 2*ones(1,N);
bnet = mk_bnet(dag, node_sizes, 'names', {'B','E','A','J','M'});
draw_graph(bnet.dag)
bnet.CPD{B} = tabular_CPD(bnet, B, [.999, .001]);
bnet.CPD{E} = tabular_CPD(bnet, E, [.998, .002]);
bnet.CPD{A} = tabular_CPD(bnet, A, [.999, .06, .71, .05, .001, .94, .29, .95]);
bnet.CPD{J} = tabular_CPD(bnet, J, [.95, .10, .05, .90]);
bnet.CPD{M} = tabular_CPD(bnet, M, [.99, .30, .01, .70]);
engine = jtree_inf_engine(bnet);
%    the conditional Prob P(J|M,B)
evidence = cell(1,N);
evidence{M} = 2;
evidence{B} = 2;
```

```
[engine, loglik] = enter_evidence(engine, evidence);
marg = marginal_nodes(engine, J);
p1=marg.T(2)
%    the conditional P(M|B)
evidence = cell(1,N);
evidence{B} = 2;
[engine, loglik] = enter_evidence(engine, evidence);
marg = marginal_nodes(engine, M);
p2=marg.T(2)
% the conditional Prob P(J,M|B)
p=p1*p2
```

Matlab-code for Problem 1

```
##################################
clear all
close all
ratio=0.4
noise=wavread('C:\Baysian\report\noises\exhibition.wav');
for i=1:1
        string1=int2str(i)
        string2=sprintf('C:\\Baysian\\Ying\\%s.wav',string1);
        string3=sprintf('C:\\Baysian\\Ying\\%s_noisy.wav',string1);
sample=wavread(string2);
wavplay(sample,8000)
plot(sample);
xlabel('time');
ylabel('amplitude');
title('clean data');
print    -djpeg 'C:\Baysian\Powerpoint\speech_clean.jpg'
figure;
M=length(sample);
noisy=ratio*noise(1:M)+sample(1:M);
wavplay(noisy,8000)
plot(noisy);
wavwrite(noisy,8000,string3);
xlabel('time');
ylabel('amplitude');
title('noise distorted data');
end
```

```
% Demo of wavelet-based function estimation
clear all
close all
ratio=0.05/0.11;
string2=sprintf('C:\\Baysian\\report\\Ying\\1_noisy.wav');
string3=sprintf('C:\\Baysian\\report\\Ying\\1_denoisy.wav');
  [s,fs,nbits] = wavread(string2);
plot(s);
xlabel('time');
ylabel('amplitude'); title('noisy data');
figure
number=floor(length(s)/1024)
for (count1=1:number)
% (i) Make Doppler Signal on [0,1]
t=linspace(0,1,1024);
sig = sqrt(t.*(1-t)).*sin((2*pi*1.05) ./(t+.05));
count2=(count1-1)*1024+1;
count3=(count1)*1024;
sig=s(count2:count3);
% (ii) Add noise of size 0.1. Make sure the noise is fixed
% by fixing the seed
% randn('seed',1)
% sign = sig + 0.01 * randn(size(sig));
% (iii) You may plot the noisy signal here.
figure(2)
% plot(t, sign)
% (iv) Take the filter H, in this case this is SYMMLET 4
filt = [ -0.07576571478934    -0.02963552764595    ...
                   0.49761866763246      0.80373875180522    ...
                   0.29785779560554    -0.09921954357694    ...
                 -0.01260396726226      0.03222310060407];
filt = MakeONFilterExt('Vaidyanathan',2)
% (v) Transfer the signal in the wavelet domain.
% Choose L=8, eight levels of decomposition
sw = dwtr(sig,8, filt);
% At this point you may view the sw. Is it disbalanced?
% Is it decorrelated?
%(vi) Let's now apply Bayesian Shrinkage.
```

```
swt = sw;
swt(2^5+1:end) = swt(2^5+1:end).*ratio;
% (vii) Return now thresholded object back to the time
% domain. Of course with the same filter and L.
a=idwtr(swt,8, filt);
% (viii) Check if you made a good estimate...
figure(3)
% plot(t, a, '-')
signew(count2:count3)=a;
end
plot(signew);xlabel('time');ylabel('amplitude');title('denoise data');
```