Numerical Assignment Report

Kaustav Khatua , Roll Number: 181075

Question 1: Mixture of Gaussians with cross-validation

a) Function em_estimates() calculates mean vectors, variance-covariance matrices and prior probabilities for given number of classes.

b) For cross-validation we may use LOOCV, 10 fold or 5 fold cv. 10 fold and LOOCV is computationally heavy; so we opt for 5 fold cv. Here loss function is -$\sum_{i=1}^{n} logf(x_i)$ .Where,

$$\text{f}(x_i) = \sum_{c=1}^{C} \pi_c \text{ . f}(x_i \mid (\mu_c \text{ , } \sum_c))$$

μ_c is mean vector and $\sum_c$ is variance-covariance matrix for c-th class.

```
cv_log_likelihood_values
```

```
## [1] 5.362656 5.260323 5.094372 5.108867 5.105765
```

Clearly third component is the least; ie. for C=4 loss function is minimum. So, according to cross-validation we choose **C=4**.

c)

```
aic_values
```

```
## [1] 5356.927 5319.206 5072.568 5072.785 5074.441
```

Fourth component is the least; ie. according to AIC values **C=4** is the right choice for C.

d) We can see that models chosen by Cross-Validation and AIC values are same.

We know that,

$$\text{AIC} = \text{- 2 . } \sum_{i=1}^{n} \log \text{ (f}(x_i) \text{) + 2 . np}$$

where, np is number of fitted parameters.

Generally model chosen by AIC values are better; because AIC value takes into consideration the number of parameters fitted, along with negative log likelihood value. If fitting one more class does not decrease log likelihood that much so that it can suppress increase in number of fitted parameters, AIC value will not decrease.

So, we **may say C=4** is the right model.

e) Using em_estimates() function we will now calculate the estimates.

```
estimates<-em_estimates(dat,4)
```

Mean vectors are:

$\mu_1 = \begin{pmatrix} 30.34921 \\ 9.239731 \end{pmatrix}$, $\mu_2 = \begin{pmatrix} 10.70645 \\ 3.884783 \end{pmatrix}$, $\mu_3 = \begin{pmatrix} 19.85683 \\ 4.886058 \end{pmatrix}$, $\mu_4 = \begin{pmatrix} 32.03310 \\ 6.910311 \end{pmatrix}$

Variance-Covariance matrices are:

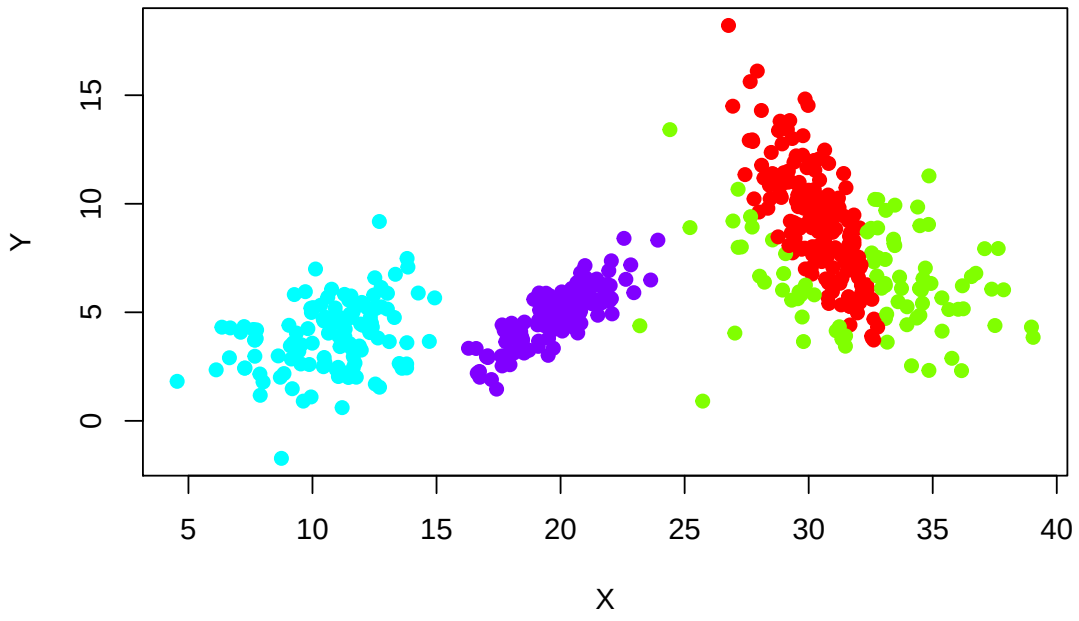
$\sum_1 = \begin{pmatrix} 2.130160 & -2.868869 \\ -2.868869 & 7.620628 \end{pmatrix}$, $\sum_2 = \begin{pmatrix} 4.203549 & 1.103098 \\ 1.103098 & 2.601429 \end{pmatrix}$, $\sum_3 = \begin{pmatrix} 2.604889 & 1.775947 \\ 1.775947 & 1.751829 \end{pmatrix}$,

$\sum_4 = \begin{pmatrix} 9.911884 & -2.149897 \\ -2.149897 & 5.178565 \end{pmatrix}$

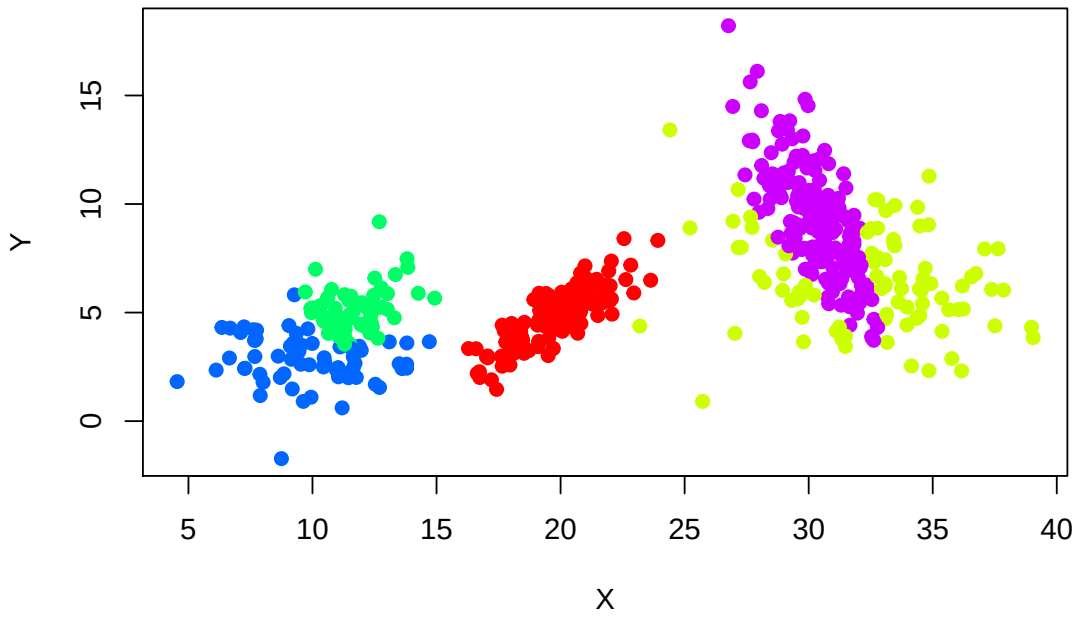
Prior probabilities are : $\pi_1 = 0.2913963$, $\pi_2 = 0.2383019$, $\pi_3 = 0.2497280$, $\pi_4 = 0.2205738$

f) Now we will plot data according to their assigned probabilities.

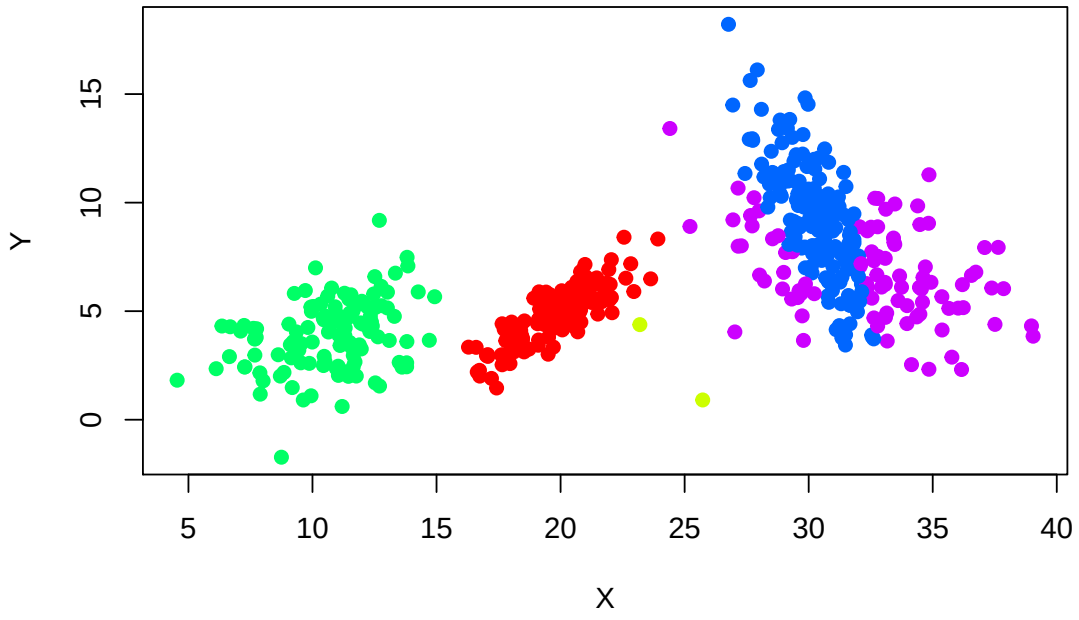
```
posterior_probabilities<-estimates$posterior_probabilities
assigned_class<-apply(posterior_probabilities,1,which.max)
palette(rainbow(4))
plot(dat,col=assigned_class,pch=19)
```



Note: I have reported the value of C according to the last run of code, where the AIC value and cross-validation methods are indicating the same model. But it was not the case all the time. Cross-validation values are more or less same, but AIC values are not stable; ie. model according to AIC values are changing, it is variying between 4 and 5 most of the times. If we choose C=5 then a class has negligible prior probability, which is indicated in the graph also.



C=5 has another problem; estimates are not stable, as a result two figures for C=5 are different.



Cross-Validation is estimating test error more efficiently,than AIC as it is considering more number of test set possible from our dataset. So, I have fitted model with C=4.

Question 2 : Bayesian Logistic Regression with MCMC

a) Posterior disribution of β given y is :

$$\pi(\beta|y) = c.exp(-\frac{\beta^T\beta}{200}) \prod_{i=1}^{n} p_i^{y_i}.(1-p_i)^{(1-y_i)}$$

where, $p_i = \frac{1}{1+exp(-x_i^T\beta)}$ and c is suitable constant such that $\pi(\beta|y)$ is a valid pdf.

b) Coefficient estimates obtained from fitting logistic regression (using glm() function) are our starting values of β. It is a good starting value as it is the range of posterior distribution and it contains information about our data.

```
logistic_regression<-glm(y~.,data=required_data[,-2],family=binomial)
logistic_regression$coefficients
```

```
##  (Intercept)           x2           x3           x4           x5
## -3.365769467  2.980802136  0.707724604  0.006799132 -1.839955659
```

These are our starting values.

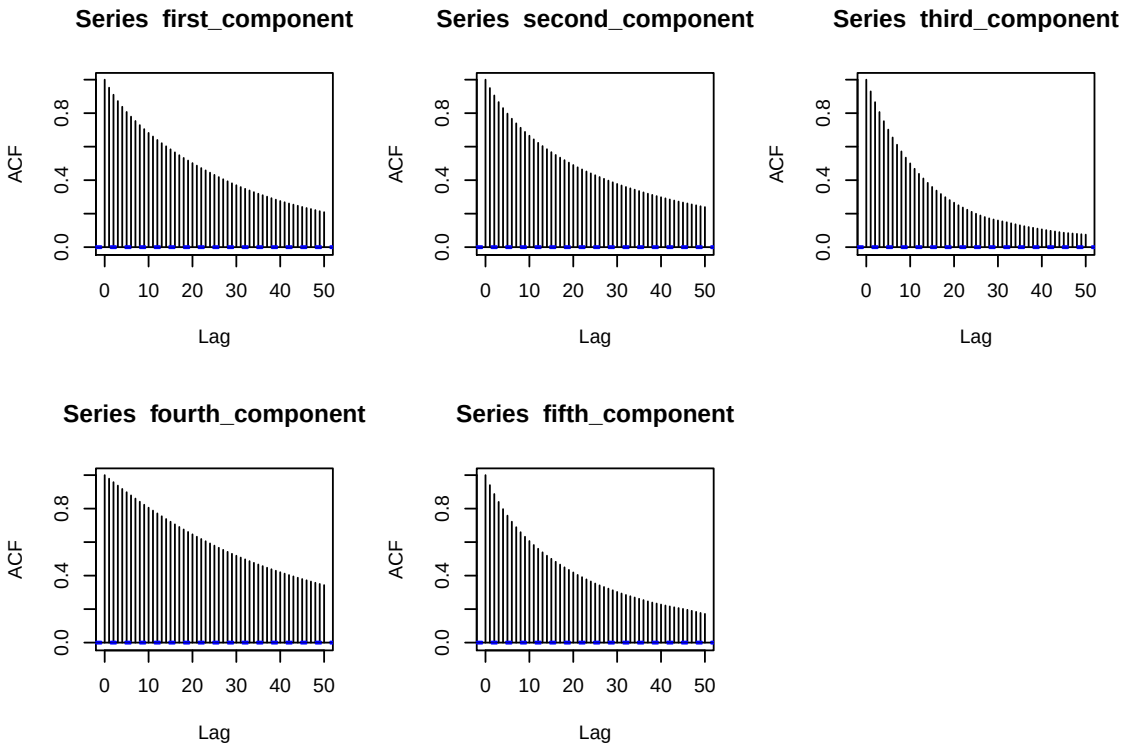
c) Our proposed distribution is N(x_t,Σ). Where Σ is a diagonal matrix with diagonal elements as standard erros obtained from using glm() function. We further tuned the elements to get acceptance probability within the range of 0.23 to 0.3.

```
summary(logistic_regression)$coefficients
```

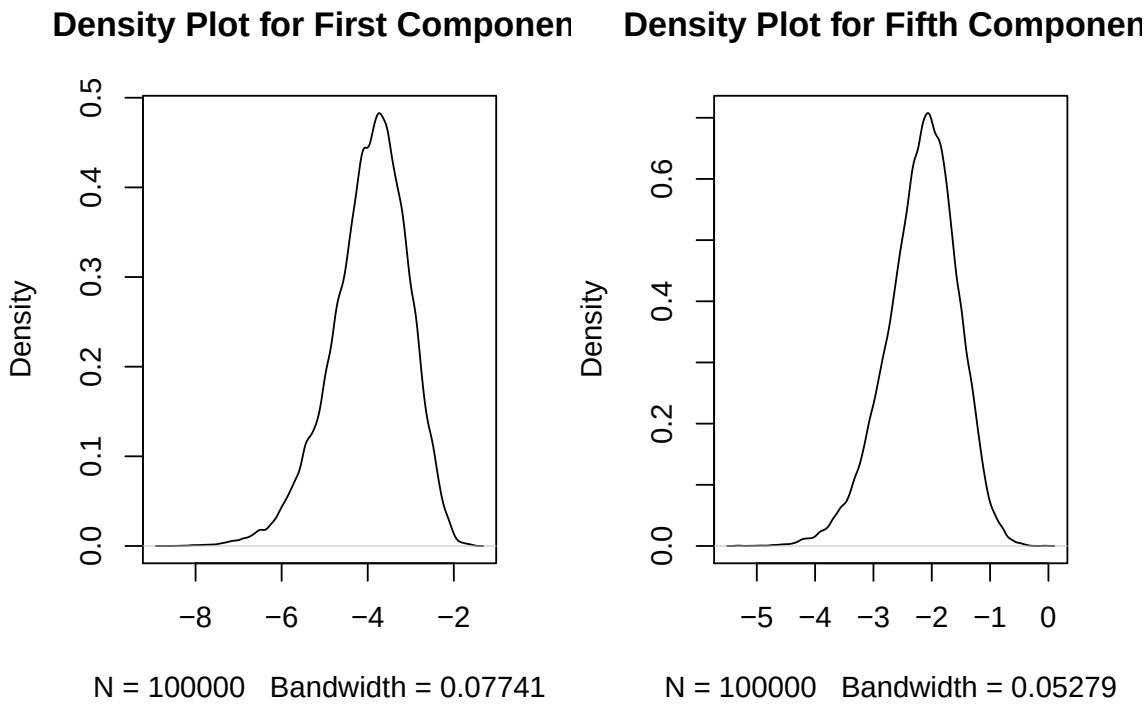
```
##                 Estimate Std. Error    z value     Pr(>|z|)
## (Intercept) -3.365769467  0.7392390 -4.5530192 5.288145e-06
## x2           2.980802136  0.7459832  3.9958036 6.447518e-05
## x3           0.707724604  0.3859694  1.8336287 6.670913e-02
## x4           0.006799132  0.4007907  0.0169643 9.864651e-01
## x5          -1.839955659  0.5209351 -3.5320247 4.123908e-04
```

Tuned h values are : 0.74, 0.743, 0.38, 0.2, 0.55 and the resultant acceptance probability is 0.23445.

d)



```
par(mfrow=c(1,2))
plot(density(first_component),main="Density Plot for First Component")
plot(density(fifth_component),main="Density Plot for Fifth Component")
```



e) Posterior mean estimates of β are :

```
beta.est
```

```
## [1] -3.965890758  3.548322811  0.825916764 -0.000234651 -2.173574938
```