HW 2 Solutions

Author:
Fateme Noorzad

StudentID:
810198271

Machine Learning
Fall 2020

Contents

Chapter 1

Question 2

1.1 Part A

Based on the given distribution we have:

$$p(D|\theta) = \prod_{i=1}^{n} p(x_i|\theta) = \prod_{i=1}^{n} \frac{1}{\theta}\mathbf{1}(0 \leq x_i \leq \theta) = \frac{1}{\theta^n}\mathbf{1}(max_i(x_i) \leq \theta)\mathbf{1}(0 \leq min_i(x_i))$$

It is clear that as θ increases, $p(D|\theta)$ decreases monotonically if $\mathbf{1}(max_i(x_i) \leq \theta)$ and $\mathbf{1}(0 \leq min_i(x_i))$ are one. Therefore, in order to maximize $p(D|\theta)$, we need to:

$$\hat{\theta} = max_i(x_i)$$

1.2 Part B

It is stated that the maximum value of the drawn samples is 0.6. Therefore, $max_i x_i = 0.6$ and for $p(D|\theta)$ we have:

$$p(D|\theta) = \begin{cases} 0 & 0 \leq \theta \leq max_i(x_i) = 0.6 \\ \frac{1}{\theta^n} & 0.6 \leq \theta \leq 1 \end{cases}$$

Therefore, the plot can be as below:

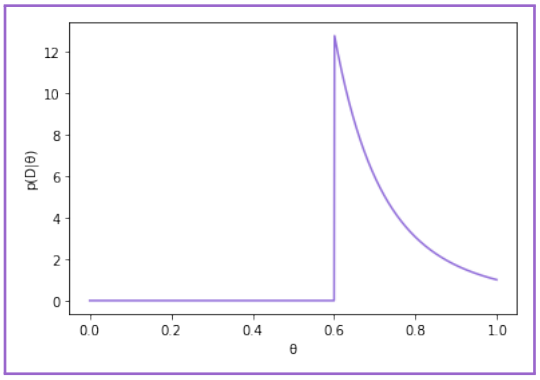

FIGURE 1.1: $p(D|\theta)$ for $0 \leq \theta \leq 1$

Chapter 2

Question 3

In this problem, it is not mentioned whether we have n samples drawn from the whole population or not. Therefore, both cases are discussed.

- **Case I: n samples are drawn**:
  As the title suggests, n samples as $\{\mathbf{x}_1,\ \mathbf{x}_2,\ ...,\ \mathbf{x}_n\}$ from the discrete distribution are drawn. The likelihood of these samples is:

$$P\{\mathbf{x}_1,\ \mathbf{x}_2,\ ...,\ \mathbf{x}_n|\boldsymbol{\theta}\} = \prod_{j=1}^{n}\prod_{i=1}^{d}\theta_i^{x_{ji}}(1-\theta_i)^{(1-x_{ji})}$$

  Based on the above likelihood function, the log-likelihood is:

$$l(\boldsymbol{\theta}) = ln(P\{\mathbf{x}_1,\ \mathbf{x}_2,\ ...,\ \mathbf{x}_n|\boldsymbol{\theta}\}) = \sum_{j=1}^{n}\sum_{i=1}^{d} x_{ji}ln(\theta_i) + (1-x_{ji})ln(1-\theta_i)$$

  Now, all we need to do is to find the maximum of $l(\boldsymbol{\theta})$, which is carried out by derivating the log-likelihood with respect to $\boldsymbol{\theta}$ as below:

$$\begin{aligned}
&\nabla_{\theta_i} l(\theta) = 0 \\
\rightarrow &\frac{1}{\theta_i}\sum_{j=1}^{n} x_{ji}\frac{1}{1-\theta_i}\sum_{j=1}^{n}(1-x_{ji}) = 0 \\
\rightarrow &(1-\hat{\theta}_i)\sum_{j=1}^{n} x_{ji} = \hat{\theta}_i\sum_{j=1}^{n}(1-x_{ji}) \\
\rightarrow &\hat{\theta}_i = \frac{1}{n}\sum_{j=1}^{n} x_{ji} \\
\rightarrow &\hat{\boldsymbol{\theta}} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_j
\end{aligned}$$

  Therefore, in this case, the estimation of $\boldsymbol{\theta}$ is the sample mean.

- **Case II: one samples is drawn**:
  As the above calculations suggest, in the case where we only have one

sample, the etimation turns out to be:

$$\begin{gathered}\nabla_{\theta_i} l(\theta) = 0 \\ \rightarrow \frac{1}{\theta_i} x_i + \frac{1}{1-\theta_i}(1-x_i) = 0 \\ \rightarrow (1-\hat{\theta}_i)x_i = \hat{\theta}_i(1-x_i) \\ \rightarrow \hat{\theta}_i = x_i \\ \rightarrow \hat{\boldsymbol{\theta}} = \mathbf{x}\end{gathered}$$

Chapter 3

Question 4

3.1 Part A

Assuming $p(x|\mu) = N(\mu, \sigma^2)$ holds. However, in the given distribution, μ is not known, all we know about it is that $p(\mu) = N(\mu_0, \sigma_0^2)$ and the values of μ_0, σ_0^2. Therefore, on a given dataset as D, by employing the Bayesina rule, we get:

$$p(\mu|D) = \frac{p(D|\mu)p(\mu)}{p(D)} = \alpha p(D|\mu)p(\mu)$$

Under the assumption that the samples of the set are i.i.d we get:

$$p(\mu|D) = \alpha \prod_{i=1}^{n} p\{x_i|\mu\}p(\mu) = \alpha \prod_{i=1}^{n} [\frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(x_i-\mu)^2}{2\sigma^2})][\frac{1}{\sqrt{2\pi\sigma_0^2}} exp(-\frac{(\mu-\mu_0)^2}{2\sigma_0^2})]$$

$$= \frac{1}{(2\pi)^{\frac{n}{2}+1}\sigma^n\sigma_0} exp(-\sum_{i=1}^{n}[\frac{(x_i-\mu)^2}{2\sigma^2}] - \frac{(\mu-\mu_0)^2}{2\sigma_0^2}) \rightarrow \frac{\mu^2 - 2\mu\frac{\sigma_0^2\sum_{i=1}^{n} x_i + \sigma^2\mu_0}{\sigma_0^2 n + \sigma^2}}{\frac{2\sigma_0^2\sigma^2}{\sigma_0^2 n + \sigma^2}}$$

$$\rightarrow \mu_n = \frac{\sigma_0^2\sum_{i=1}^{n} x_i + \sigma^2\mu_0}{\sigma_0^2 n + \sigma^2},\ \sigma_n^2 = \frac{2\sigma_0^2\sigma^2}{\sigma_0^2 n + \sigma^2}$$

$$\text{if } n_0 = \frac{\sigma^2}{\sigma_0^2} : \boxed{\mu_n = \frac{1}{n+n_0}\sum_{i=-n_0+1}^{n} x_i} \quad \boxed{\sigma_n^2 = \frac{\sigma^2}{n+n_0}}$$

3.2 Part B

As we know, the class-conditional density is calculated based on $p(x|D) = N(\mu_n, \sigma^2 + \sigma_n^2)$. Therefore, in order to achieve the desired distribution, based on the above formula, the values of μ_0, σ_0^2 needs to be picked accordingly. In addition, woth suitable choices of these values, Bayesian learning can be interpreted as ML learning with:

$$\mu_0 = \frac{1}{n_0}\sum_{i=-n_0+1}^{0} x_i,\ \sigma_0^2 = \frac{\sigma^2}{n_0}$$

which both of these values are calculated based on the first n_0 observed samples.

Chapter 4

Question 5

In order to find the MAP estimation, we need to do the following:

$$\hat{\mu}_{\text{MAP}} = argmax\{p(\mathbf{X}|\mu)p(\mu)\}$$

Now, all we need to do is to calculate the likelihood function:

$$p(\mathbf{X}|\mu)p(\mu) = \prod_{i=1}^{n} p(\mathbf{x}_i|\mu)p(\mu) = \prod_{i=1}^{n}[\frac{1}{\sqrt{2\pi\sigma^2}}exp(-\frac{(\mathbf{x}_i-\mu)^2}{2\sigma^2})][\frac{\mu}{\sigma_\mu^2}exp(-\frac{\mu^2}{2\sigma_\mu^2})]$$
$$= \frac{\mu}{(2\pi)^{\frac{n}{2}}\sigma^n\sigma_\mu}exp(-\sum_{i=1}^{n}[\frac{(\mathbf{x}_i-\mu)^2}{2\sigma^2}]-\frac{\mu^2}{2\sigma_\mu^2})$$

As the next step, the log-likelihood value is calculated:

$$l(\mu) = ln(\mu) - ln((2\pi)^{\frac{n}{2}}\sigma^n\sigma_\mu) - \sum_{i=1}^{n}[\frac{(\mathbf{x}_i-\mu)^2}{2\sigma^2}]-\frac{\mu^2}{2\sigma_\mu^2}$$

Afterward, in order to find the argument that maximizes the above equation, the derivation of the stated equation with respect to μ is calculated.

$$\frac{\partial}{\partial\mu}l(\mu) = 0$$

$$\rightarrow \frac{1}{\mu} + \sum_{i=1}^{n}[\frac{\mathbf{x}_i-\mu}{\sigma^2}] - \frac{\mu}{\sigma_\mu^2} = 0$$

$$\xrightarrow{(\text{if } \mu\neq 0)} \mu^2(\frac{n\sigma_\mu^2+\sigma^2}{\sigma_\mu^2\sigma^2}) - \frac{\mu}{\sigma^2}\sum_{i=1}^{n}\mathbf{x}_i - 1 = 0$$

$$\rightarrow \hat{\mu}_{\text{MAP}} = \frac{\frac{1}{\sigma^2}\sum_{i=1}^{n}\mathbf{x}_i \pm \sqrt{(\frac{1}{\sigma^2}\sum_{i=1}^{n}\mathbf{x}_i)^2 + \frac{n\sigma_\mu^2+\sigma^2}{\sigma_\mu^2\sigma^2}}}{2\frac{n\sigma_\mu^2+\sigma^2}{\sigma_\mu^2\sigma^2}}$$

Chapter 5

Question 6

Suppose we have K Poisson distribution with parameters λ_1, ... λ_K mixed together with proportions α_1, ... α_K. Therefore, the likelihood of this distribution can be written as below:

$$L\{\boldsymbol{\lambda}|\mathbf{X}\} = p\{D|\lambda\} = \prod_{i=1}^{n} p\{\mathbf{x}_i|\lambda\}$$

Therefore, the log-likelihood can be represented as below:

$$l(\boldsymbol{\lambda}) = \sum_{i=1}^{N} ln\{\sum_{j=1}^{K} \alpha_j P_j(\mathbf{x}_i|\lambda_j)\}$$

However, we cannot obtain analytical solution for $\boldsymbol{\lambda}$ by simply setting the serivative of the LL obtained above to zero because of the logarithm on sum. Therefore, we employ Jensen's inequality. Therefore, $Q(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{(g)})$ is:

$$Q(\boldsymbol{\lambda}, \boldsymbol{\lambda}^{(g)}) = \sum_{j=1}^{K}\sum_{i=1}^{N} ln(\alpha_j) p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\} + \sum_{j=1}^{K}\sum_{i=1}^{N} ln(p_j(\mathbf{x}_i|\lambda j)) p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\}$$

Now, in order to determine α_j, we used the above equation with a constrain on the fact that its sum is one. Therefore, by employing lagrangian multiplier, we get:

$$\hat{\alpha}_j = \frac{1}{n}\sum_{i=1}^{n} p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\},\ p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\} = \frac{\alpha_j^{(g)} p_j(\mathbf{x}_i|\lambda_j^{(g)})}{\sum_{t=1}^{K} \alpha_t^{(g)} p_t(\mathbf{x}_i|\lambda_t^{(g)})}$$

Afterward, we need to evaluate an estimation of λ_j, we need to create its likelihood function:

$$p_j(\mathbf{x}_i|\lambda_j) = \frac{\lambda_j^{\mathbf{x}_i} exp(-\lambda_j)}{\mathbf{x}_i!}$$

For simplicity, the log-likelihood function is created as below:

$$ln\{p_j(\mathbf{x}_i|\lambda_j)\} = \mathbf{x}_i ln(\lambda_j) - \lambda_j - ln(\mathbf{x}_i!)$$

Based on the above resutls the likelihood function for $\boldsymbol{\lambda}$ can be determined as below:

$$L(\boldsymbol{\lambda}) = \sum_{j=1}^{K}\sum_{i=1}^{N} ln(p_j(\mathbf{x}_i|\lambda j)) p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\} = \sum_{j=1}^{K}\sum_{i=1}^{N} (\mathbf{x}_i ln(\lambda_j) - \lambda_j - ln(\mathbf{x}_i!)) p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\}$$

Now as the last step, we need to derivate the above function with respect to λ_j:

$$\frac{\partial}{\partial \lambda_j} L(\lambda_j) = 0$$

$$\rightarrow \sum_{i=1}^{N} (\frac{\mathbf{x}_i}{\lambda_j} - 1) p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\} = 0$$

$$\rightarrow \hat{\lambda}_j = \frac{\sum_{i=1}^{N} \mathbf{x}_i p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\}}{\sum_{i=1}^{N} p\{j|\mathbf{x}_i, \boldsymbol{\lambda}^{(g)}\}}$$

Chapter 6

Question 7

First of all, the images are read and stored in a list. Then, their RGB average is calculated. Note that since we need the R and B class, the G average is removed.

This problem is ismulated in two ways. In the first one, GMM is fitted to the whole dataset. In the second method, GMM is fitted to each class.

6.1 Whole Dataset

6.1.1 Part A

In this part, by employing *"GaussianMixture"* from *"SKlearn"* library and setting the number of components to 2, as asked, two Gaussians create a mixture model and are fitted to the given images. The mean and covariance value which are a result of this model are as below:

	Component 1	Component 2
Mean	[106.81314379 91.7794563]	[73.22719697 62.12078031]
Covariance	[[750.57013865 268.08102344] [268.08102344 973.98691004]]	[[400.13701231 131.96282572] [131.96282572 308.07777157]]

FIGURE 6.1: Table of Mean and Variance of GMM with Two Components

Now, the contours and ellipses related to the above values for mean and covariance matrix as well as data are ploted. The results are depicted as below:

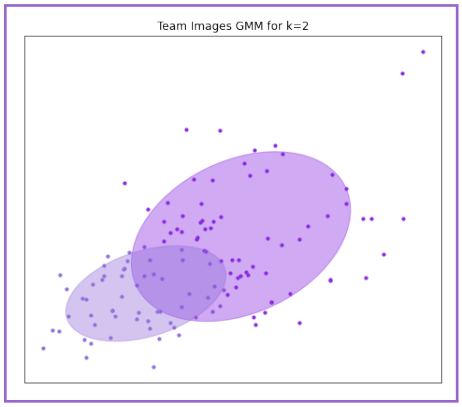

FIGURE 6.2: Ellipses of Team Images for k=2

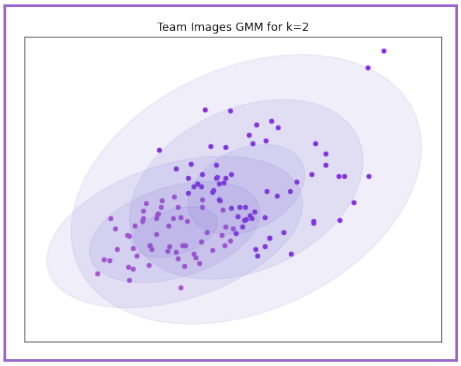

FIGURE 6.3: Contours of Team Images for k=2

6.1.2 Part B

The last part is repeated for various values of k. The results are depicted as below:

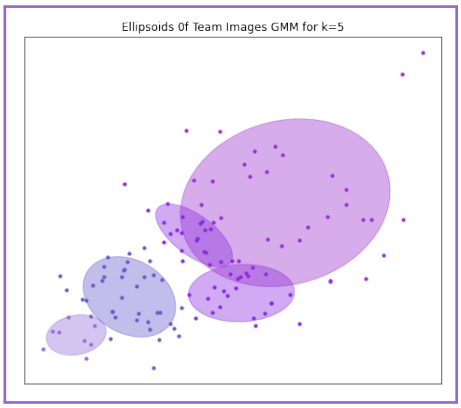

FIGURE 6.4: Ellipses of Team Images for k=5

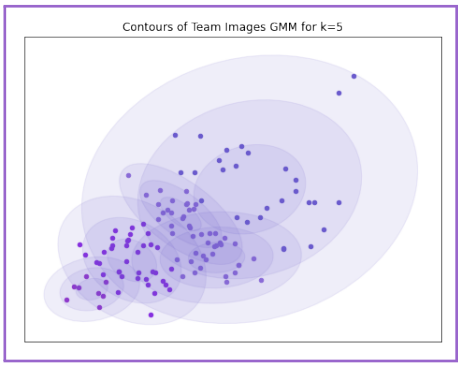

FIGURE 6.5: Contours of Team Images for k=5

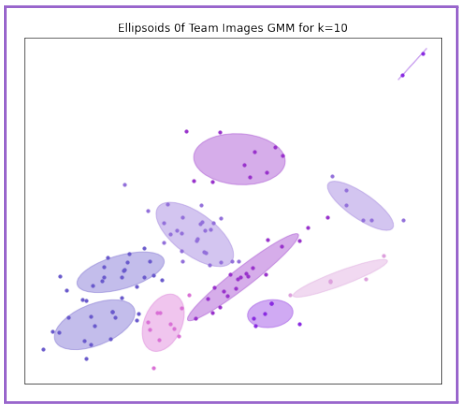

FIGURE 6.6: Ellipses of Team Images for k=10

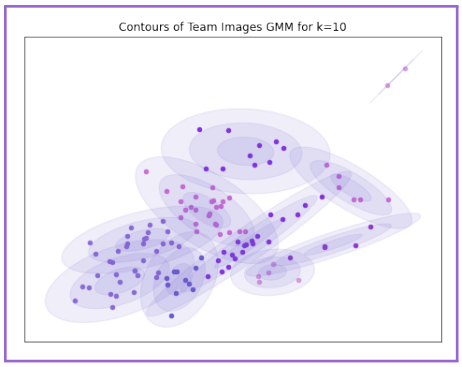

FIGURE 6.7: Contours of Team Images for k=10

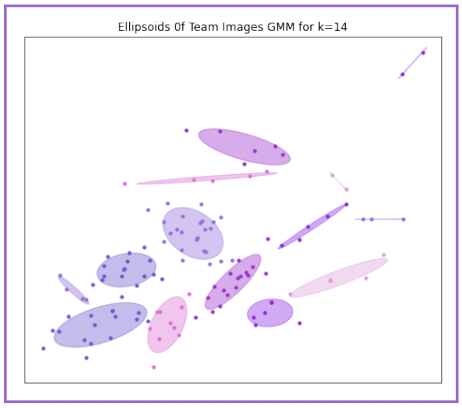

FIGURE 6.8: Ellipses of Team Images for k=14

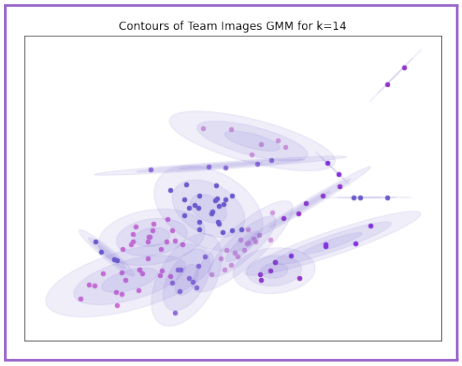

FIGURE 6.9: Contours of Team Images for k=14

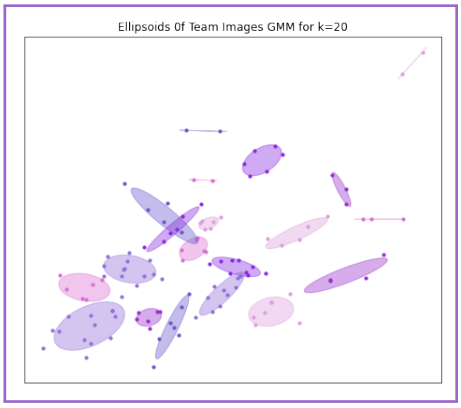

FIGURE 6.10: Ellipses of Team Images for k=20

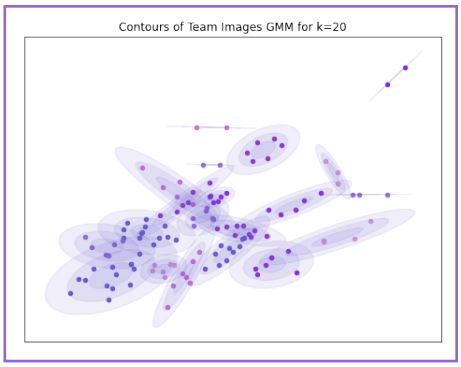

FIGURE 6.11: Contours of Team Images for k=20

In order to determine the optimal value for the number of components needed to fit the teams model, the AIC and BIC plots are depicted as below:

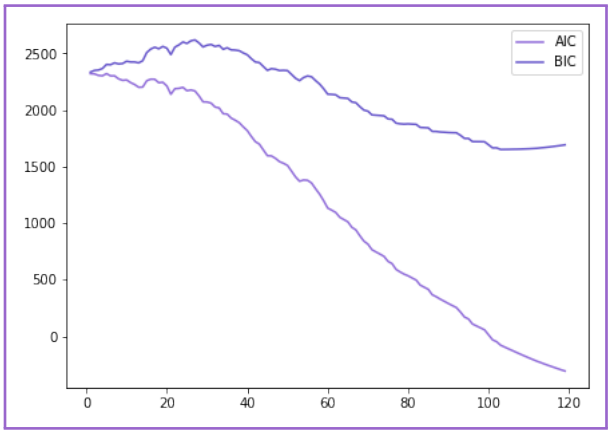

FIGURE 6.12: AIC and BIC Curves

To use AIC for model selection, we simply need to choose the model giving the smallest AIC over the set of created models. It is also crystal clear that the penalty for AIC is less than for BIC. This causes AIC to pick more complex models. Importantly, the derivation of BIC under the Bayesian probability framework means that if a selection of candidate models includes a true model for the dataset, then the probability that BIC will select the true models increases with the size of the training dataset. This fact is not true for AIC. However, a downside of BIC for smaller, less representative training datsets, it is more likely to choose models that are too simple. Therfore, a comparision of both of these measurements can give

clearer selection of optimal number of components. Note that in the selection of the number of components it is essential to note that how much cost it has for us as well.

6.2 Each Class Separated

In this method, after the images are read and their RGB average is calculated, they are separeted by their labele. For the labels with not accurate names, a name is picked based on the observations. Then the same above steps are repeated.

6.2.1 Part A

In this part, by employing *"GaussianMixture"* from *"SKlearn"* library and setting the number of components to 2, as asked, two Gaussians create a mixture model and are fitted to the given images. The mean and covariance value which are a result of this model are as below:

Manchester United	Component 1	Component 2
Mean	[97.66912718 61.47298717]	[147.967202 106.51672625]
Covariance	[[632.17809487 302.09403659] [302.09403659 323.48638795]]	[[60.9736979 -62.82591814] [-62.82591814 131.3730454]]

FIGURE 6.13: Table of Mean and Variance of GMM with Two Components for Manchester United

Chelsea	Component 1	Component 2
Mean	[102.86599546 111.80320728]	[73.99478251 81.61180176]
Covariance	[[895.04542828 667.45696758] [667.45696758 1518.52618434]]	[[413.21399756 426.54534735] [426.54534735 555.19558077]]

FIGURE 6.14: Table of Mean and Variance of GMM with Two Components for Chelsea

Now, the contours and ellipses related to the above values for mean and covariance matrix as well as data are ploted. The results are depicted as below:

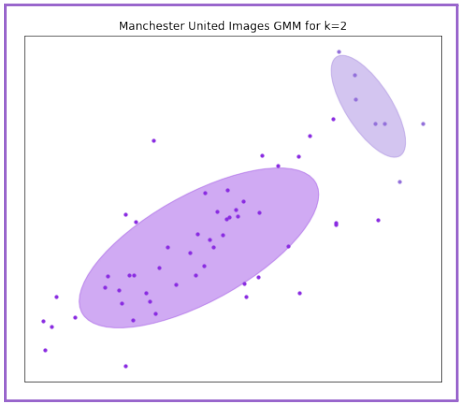

FIGURE 6.15: Ellipses of Manchester United for k=2

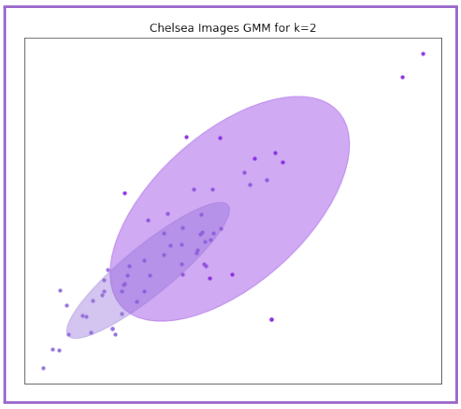

FIGURE 6.16: Ellipses of Chelsea for k=2

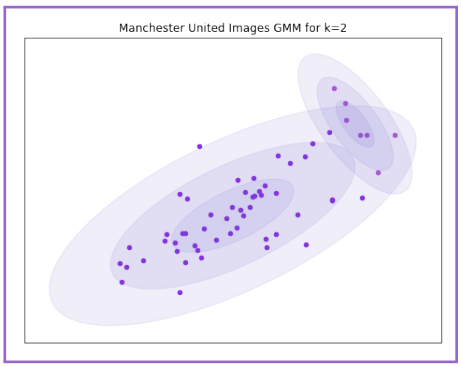

FIGURE 6.17: Contours of Manchester United for k=2

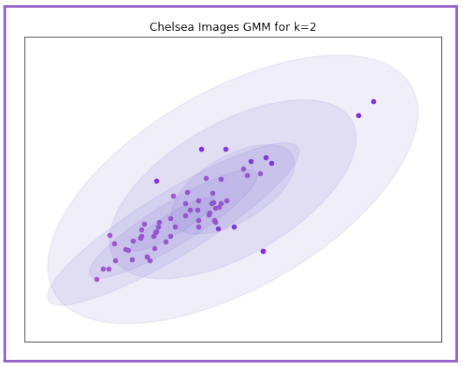

FIGURE 6.18: Contours of Chelsea for k=2

6.2.2 Part B

The last part is repeated for various values of k. The results are depicted as below:

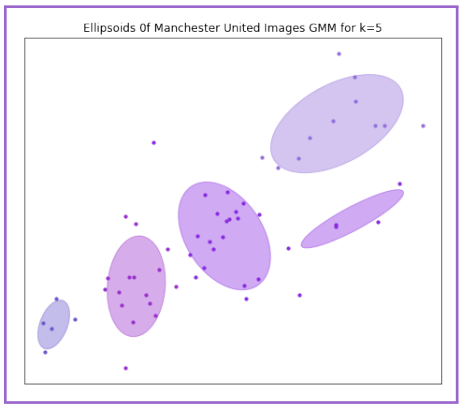

FIGURE 6.19: Ellipses of Manchester United for k=5

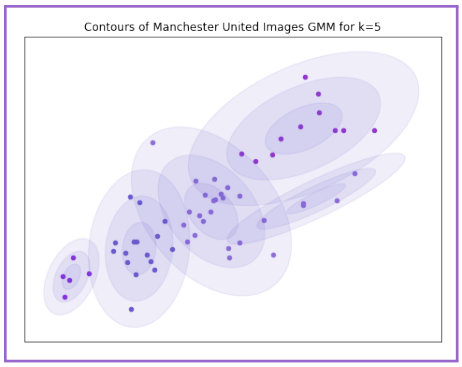

FIGURE 6.20: Contours of Manchester United for k=5

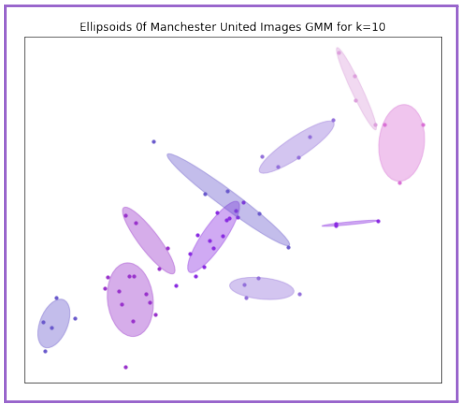

FIGURE 6.21: Ellipses of Manchester United for k=10

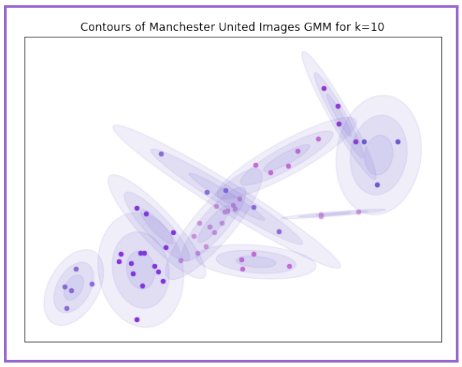

FIGURE 6.22: Contours of Manchester United for k=10

FIGURE 6.23: Ellipses of Manchester United for k=14

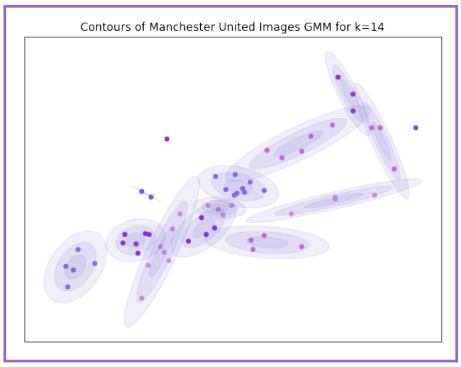

FIGURE 6.24: Contours of Manchester United for k=14

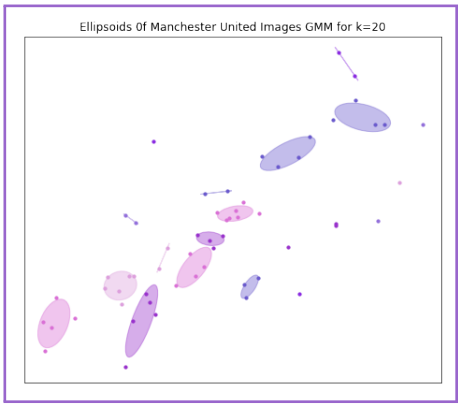

FIGURE 6.25: Ellipses of Manchester United for k=20

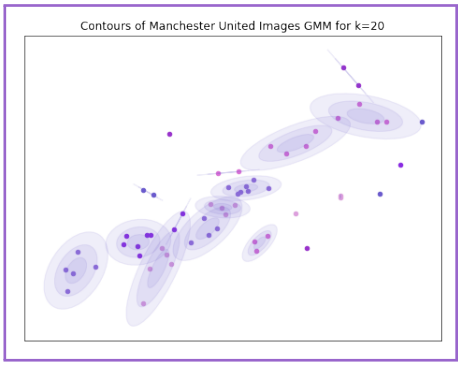

FIGURE 6.26: Contours of Manchester United for k=20

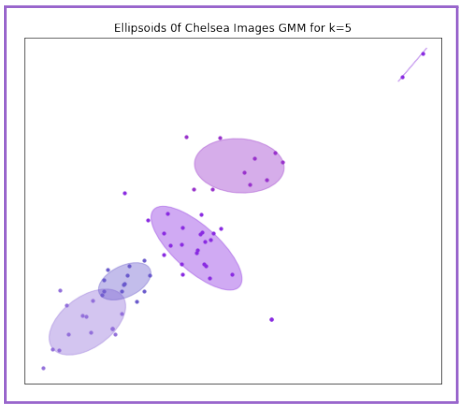

FIGURE 6.27: Ellipses of Chelsea for k=5

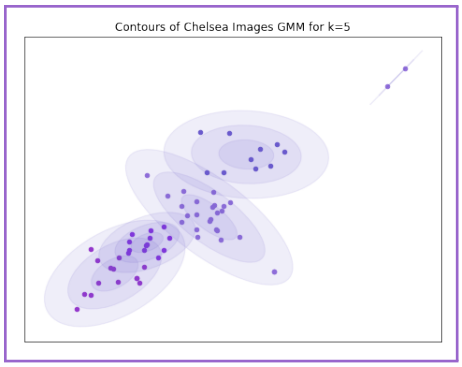

FIGURE 6.28: Contours of Chelsea for k=5

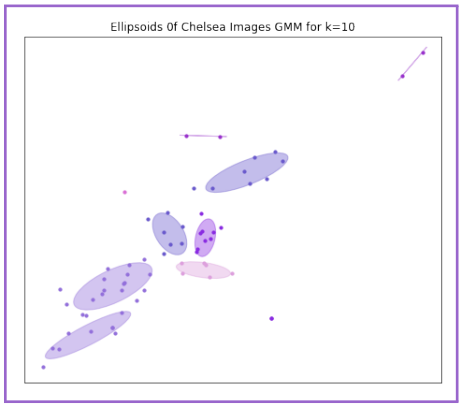

FIGURE 6.29: Ellipses of Chelsea for k=10

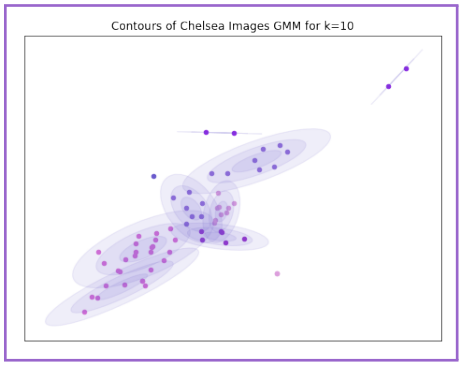

FIGURE 6.30: Contours of Chelsea United for k=10

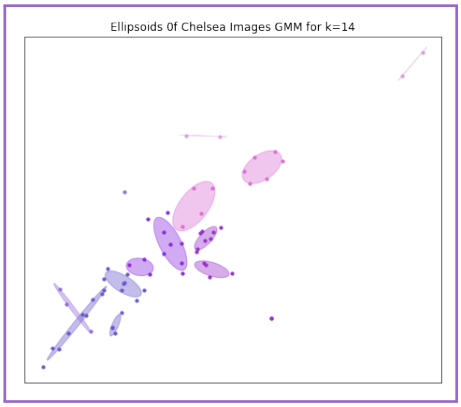

FIGURE 6.31: Ellipses of Chelsea for k=14

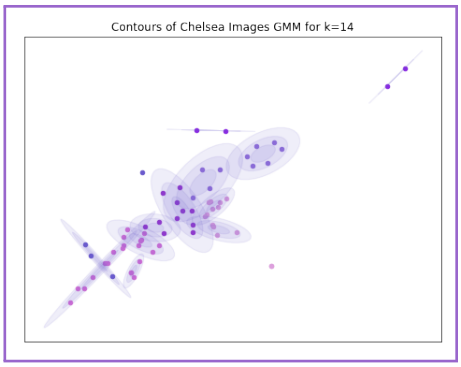

FIGURE 6.32: Contours of Chelsea for k=14

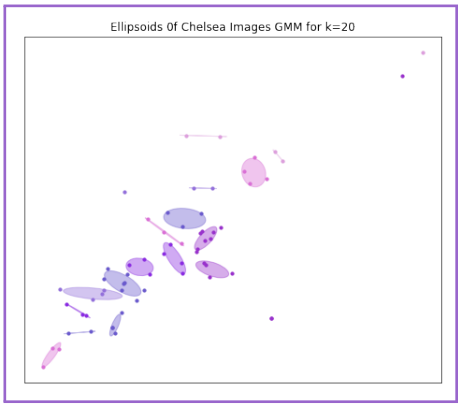

FIGURE 6.33: Ellipses of Chelsea for k=20

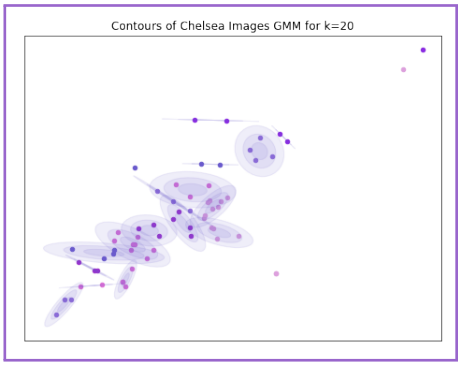

FIGURE 6.34: Contours of Chelsea for k=20

In order to determine the optimal value for the number of components needed to fit the teams model, the AIC and BIC plots are depicted as below:

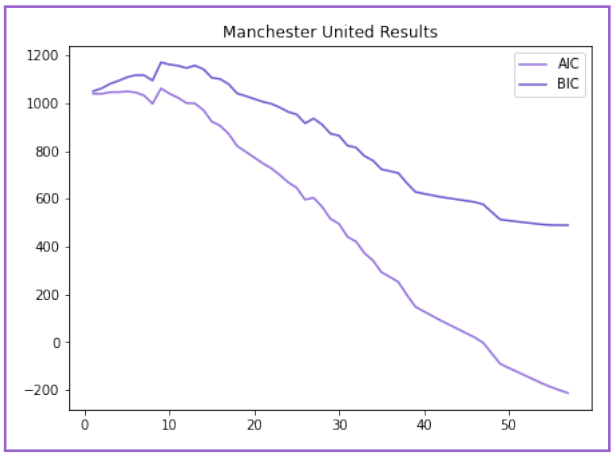

FIGURE 6.35: AIC and BIC Curves fro Manchester United

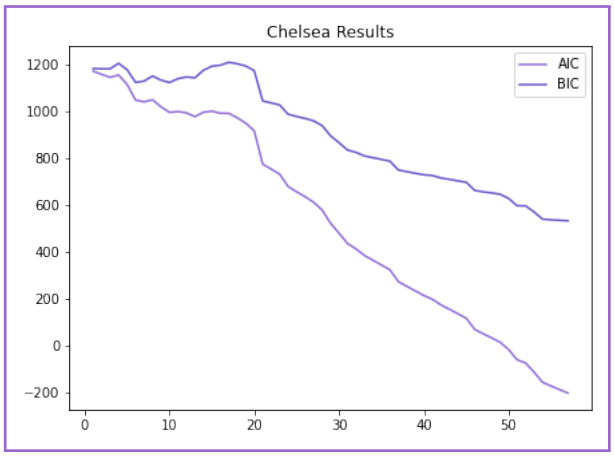

FIGURE 6.36: AIC and BIC Curves for Chlesea

The same story discussed above, holds for this case as well.

Chapter 7

Question 8

In order to gain insight of this dataset actually is, it is showns as below:

	species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE
...	...	...	...	...	...	...	...
339	Gentoo	Biscoe	NaN	NaN	NaN	NaN	NaN
340	Gentoo	Biscoe	46.8	14.3	215.0	4850.0	FEMALE
341	Gentoo	Biscoe	50.4	15.7	222.0	5750.0	MALE
342	Gentoo	Biscoe	45.2	14.8	212.0	5200.0	FEMALE
343	Gentoo	Biscoe	49.9	16.1	213.0	5400.0	MALE

FIGURE 7.1: Penguin Dataset

As the above picture shows, there are some rows with missing values. (In addition by fiding the NaN values the column with these missing datas are also shown in the code file) Therefore, we need to clean the dataset before moving forward. To do so, we have to options: remove the missing rows compeletly or interpolate them with the known values. To make a choice between the two ways of doing so, we need to check what happens to the dataset if we remove the missing rows.

```
Dataset's shape wihtout dropping the missing data is: (344, 7)
After eliminating the missing data, dataset's shape is: (334, 7)
```

FIGURE 7.2: Dataset's Shape Before and After Removing Missing Data

Since the number of rows with missing ones is not large enough to choose the second method of cleaning, the rows with NaN data are removed. Now, we have a clean dataset and can work with!

7.1 Part A

Based on the given features, the scattering plots are depicted as below:

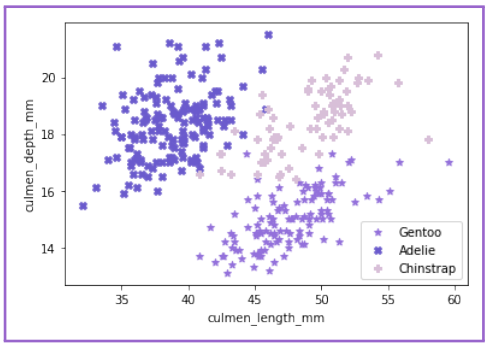

FIGURE 7.3: Dataset Based on Culmen Length mm and Culmen Depth mm

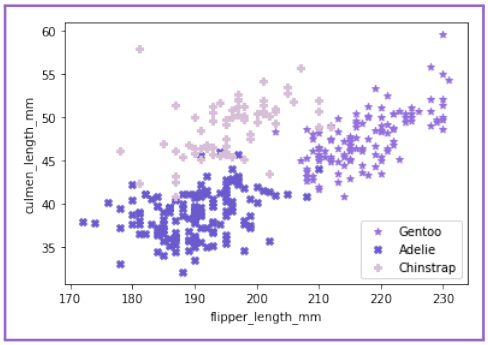

FIGURE 7.4: Dataset Based on Flipper Length mm and Culmen Length mm

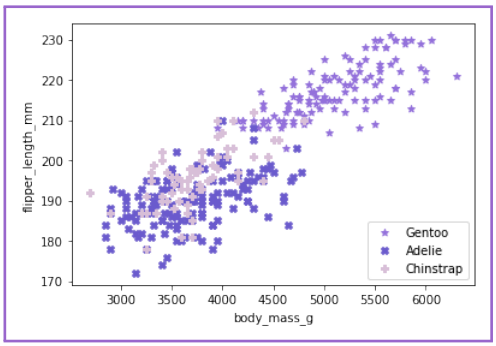

FIGURE 7.5: Dataset Based on Body Mass G and Flipper Length mm

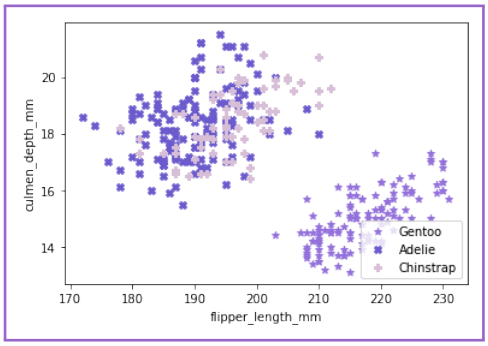

FIGURE 7.6: Dataset Based on Flipper Length mm and Culmen Depth mm

Based on the above plots we can conclude that the features used in the first plot can give a better and more accurace modeling for the given dataset. Although the second pair of features are good too, it seem that data in the first plot is more

separeted.

7.2 Part B and C

As we are asked, for each pair of features and for each class, the fitted GMM contours as well as dataset itself is depicted. (Here based on the assumption of the problem the number of Gaussina mixture models is 1)

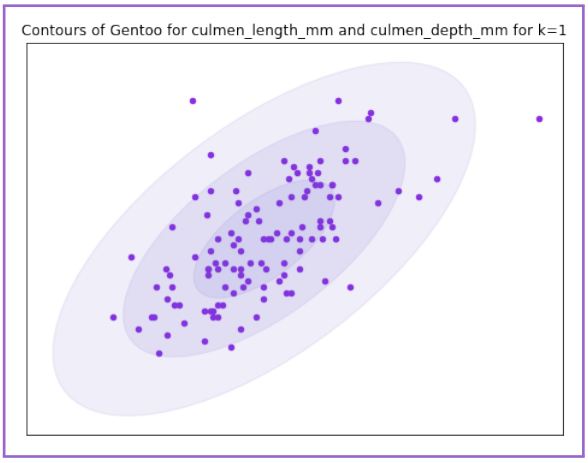

FIGURE 7.7: Contours of Gentoo Based on Culmen Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[47.5425 15.0025]
Covariance	[[9.56494475 1.95181042] [1.95181042 0.96007808]]

FIGURE 7.8: Table of Mean and Covariance of Gentoo Based on Culmen Length mm and Culmen Depth mm for k=1

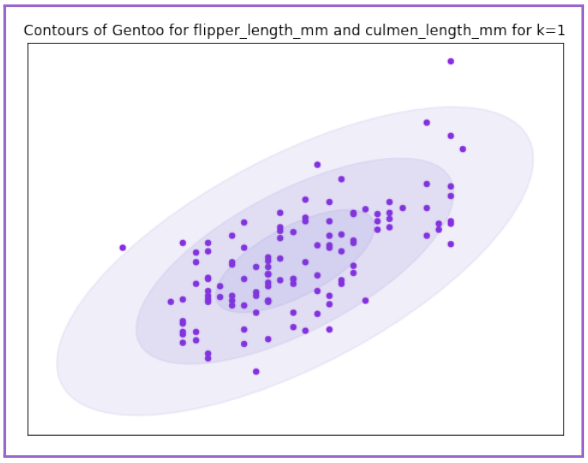

FIGURE 7.9: Contours of Gentoo Based on Flipper Length mm and Culmen Length mm for k=1

	Component 1
Mean	[217.23333333 47.5425]
Covariance	[[42.64555656 13.36591667] [13.36591667 9.56494475]]

FIGURE 7.10: Table of Mean and Covariance of Gentoo Based on Flipper Length mm and Culmen Length mm for k=1

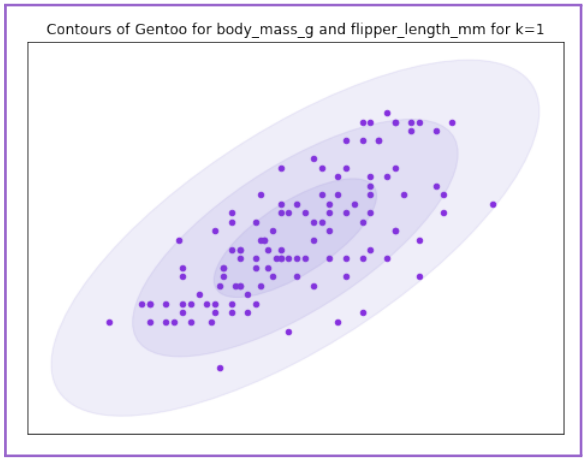

FIGURE 7.11: Contours of Gentoo Based on Body Mass G and Flipper Length mm for k=1

	Component 1
Mean	[5090.625 217.23333333]
Covariance	[[2.47677734e+05 2.31031250e+03] [2.31031250e+03 4.26455566e+01]]

FIGURE 7.12: Table of Mean and Covariance of Gentoo Based on Body Mass G and Flipper Length mm for k=1

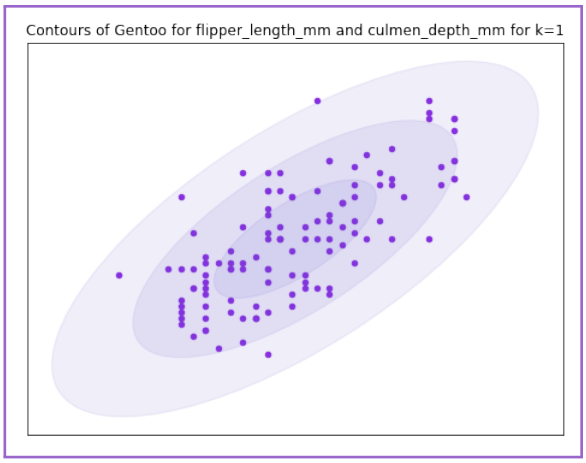

FIGURE 7.13: Contours of Gentoo Based on Flipper Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[217.23333333 15.0025]
Covariance	[[42.64555656 4.53608333] [4.53608333 0.96007808]]

FIGURE 7.14: Table of Mean and Covariance of Gentoo Based on Flipper Length mm and Culmen Depth mm for k=1

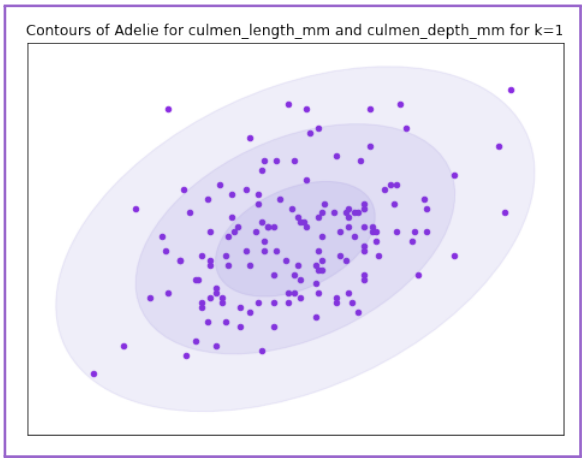

FIGURE 7.15: Contours of Adelie Based on Culmen Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[38.8239726 18.34726027]
Covariance	[[7.04086467 1.24400403] [1.24400403 1.47660308]]

FIGURE 7.16: Table of Mean and Covariance of Adelie Based on Culmen Length mm and Culmen Depth mm for k=1

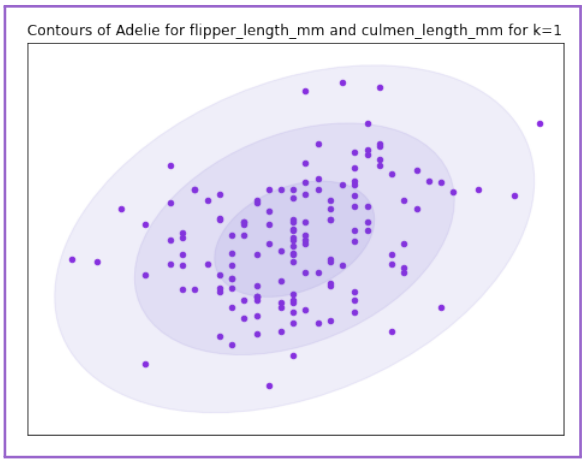

FIGURE 7.17: Contours of Adelie Based on Flipper Length mm and Culmen Length mm for k=1

	Component 1
Mean	[190.10273973 38.8239726]
Covariance	[[42.24287021 5.73041377] [5.73041377 7.04086467]]

FIGURE 7.18: Table of Mean and Covariance of Adelie Based on Flipper Length mm and Culmen Length mm for k=1

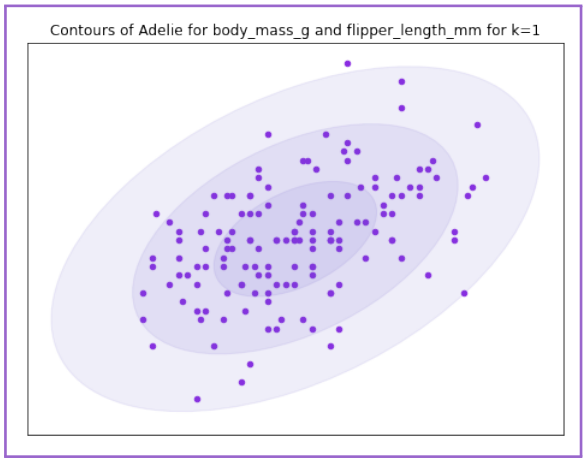

FIGURE 7.19: Contours of Adelie Based on Body Mass G and Flipper Length mm for k=1

	Component 1
Mean	[3706.16438356 190.10273973]
Covariance	[[2.08891795e+05 1.38087352e+03] [1.38087352e+03 4.22428702e+01]]

FIGURE 7.20: Table of Mean and Covariance of Adelie Based on Body Mass G and Flipper Length mm for k=1

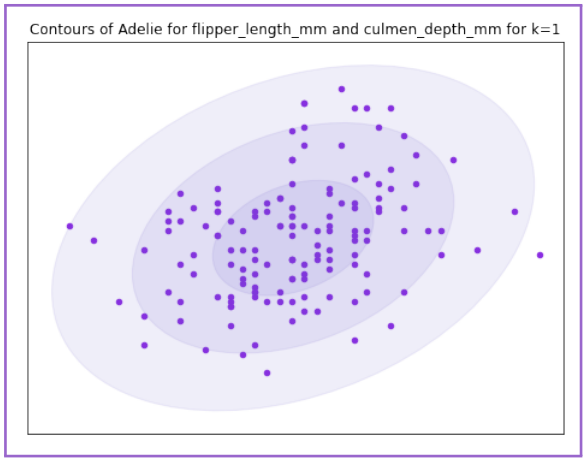

FIGURE 7.21: Contours of Adelie Based on Flipper Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[190.10273973 18.34726027]
Covariance	[[42.24287021 2.45541847] [2.45541847 1.47660308]]

FIGURE 7.22: Table of Mean and Covariance of Adelie Based on Flipper Length mm and Culmen Length mm for k=1

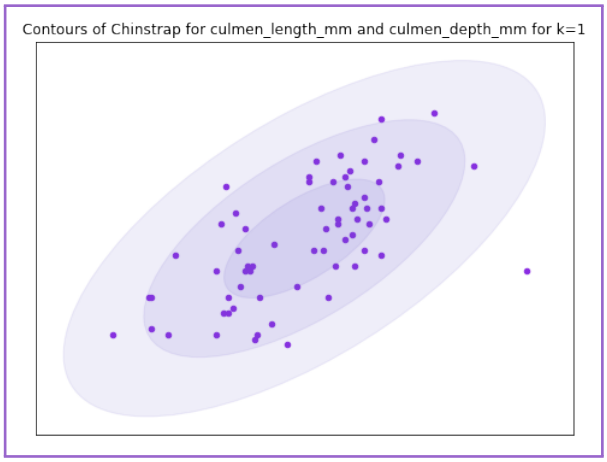

FIGURE 7.23: Contours of Chinstrap Based on Culmen Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[48.83382353 18.42058824]
Covariance	[[10.98665109 2.44136246] [2.44136246 1.27016536]]

FIGURE 7.24: Table of Mean and Covariance of Chinstrap Based on Culmen Length mm and Culmen Depth mm for k=1

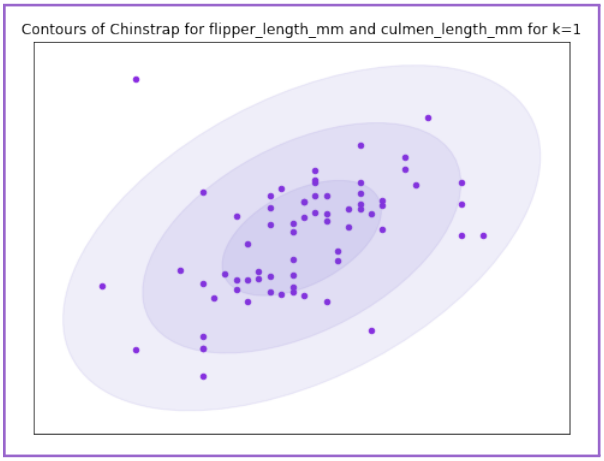

FIGURE 7.25: Contours of Chinstrap Based on Flipper Length mm and Culmen Length mm for k=1

	Component 1
Mean	[195.82352941 48.83382353]
Covariance	[[50.11591796 11.06626298] [11.06626298 10.98665109]]

FIGURE 7.26: Table of Mean and Covariance of Chinstrap Based on Flipper Length mm and Culmen Length mm for k=1

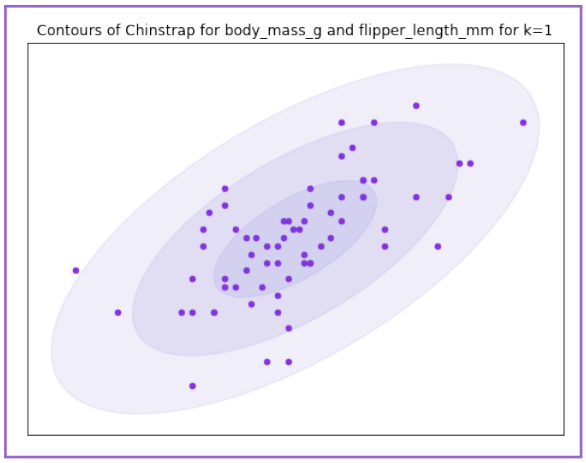

FIGURE 7.27: Contours of Chinstrap Based on Body Mass G and Flipper Length mm for k=1

	Component 1
Mean	[3733.08823529 195.82352941]
Covariance	[[1.45541198e+05 1.73267734e+03] [1.73267734e+03 5.01159180e+01]]

FIGURE 7.28: Table of Mean and Covariance of Chinstrap Based on Body Mass G and Flipper Length mm for k=1

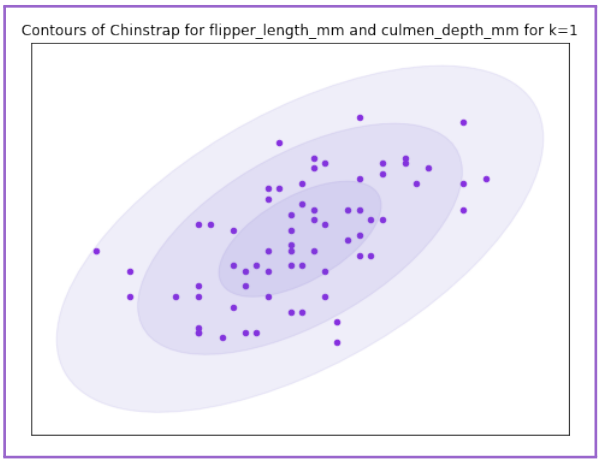

FIGURE 7.29: Contours of Chinstrap Based on Flipper Length mm and Culmen Depth mm for k=1

	Component 1
Mean	[195.82352941 18.42058824]
Covariance	[[50.11591796 4.62863322] [4.62863322 1.27016536]]

FIGURE 7.30: Table of Mean and Covariance of Chinstrap Based on Flipper Length mm and Culmen Length mm for k=1

As the next step, we are asked to report error. To do so, for each class, the log-likelihood of each pair of feature creating a model is calculated. In the end, for each class, the pair resulting in the maximum log-likelihood is picked as the best pair. The results are depicted as below:

```
The maximum value of likelihood is: -3.6786 which is for Gentoo pinguin with culmen_length_mm culmen_depth_mm and  as its selected features
The maximum value of likelihood is: -3.9280 which is for Adelie pinguin with culmen_length_mm culmen_depth_mm and  as its selected features
The maximum value of likelihood is: -3.8773 which is for Chinstrap pinguin with culmen_length_mm culmen_depth_mm and  as its selected features
```

FIGURE 7.31: Maximum of LL for Each Class

Therfore, as the above results suggest, for each class, as we stated in part a, the first pair is the best one.

7.3 Part D

By employing the best pair of features, as determined above, the resulting AIC and BIC figures are depicted as below. (Note that the scattering plot as well as the countours for all these values are depicted as well. To check them please refere to code!)

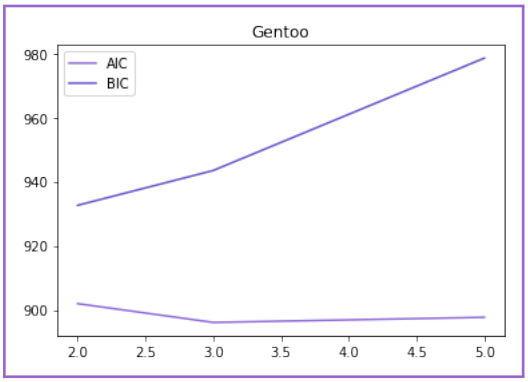

FIGURE 7.32: AIC and BIC Curves for Gentoo Based on Culmen Length mm and Culmen Depth mm

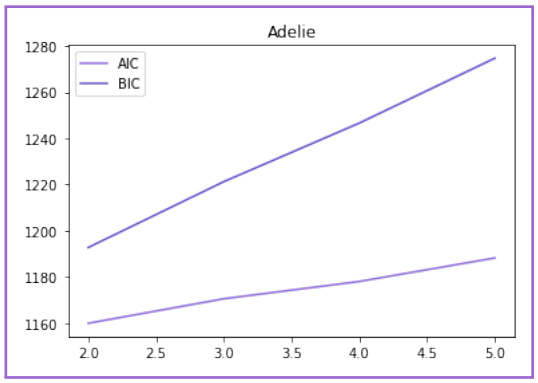

FIGURE 7.33: AIC and BIC Curves for Adelie Based on Culmen Length mm and Culmen Depth mm

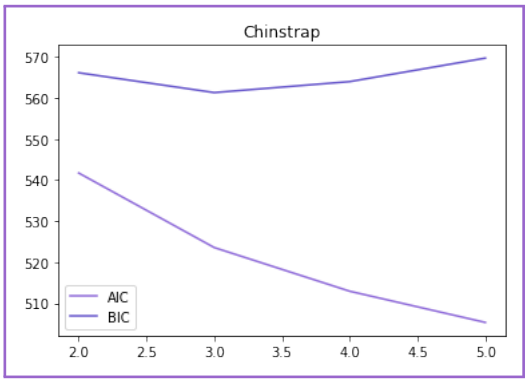

FIGURE 7.34: AIC and BIC Curves for Chinstrap Based on Culmen Length mm and Culmen Depth mm

To use AIC for model selection, we simply need to choose the model giving the smallest AIC over the set of created models. It is also crystal clear that the penalty for AIC is less than for BIC. This causes AIC to pick more complex models. Importantly, the derivation of BIC under the Bayesian probability framework means that if a selection of candidate models includes a true model for the dataset, then the probability that BIC will select the true models increases with the size of the training dataset. This fact is not true for AIC. However, a downside of BIC for smaller, less representative training datsets, it is more likely to choose models that are too simple. Therfore, a comparision of both of these measurements can give clearer selection of optimal number of components. Note that in the selection of the number of components it is essential to note that how much cost it has for us as well. In the case of this problem, between the given number of components, for *Gentoo* the optimum number is 3, while the same value for *Adelie* and *Chinstrap* is 2 and 3 respectively.

Chapter 8

Question 9

First of all, before starting to find the best Gaussian distribution describing the given database, a plot of what it actually looks like can give some insights. Therefore, after downloading it, it is depicted as below:

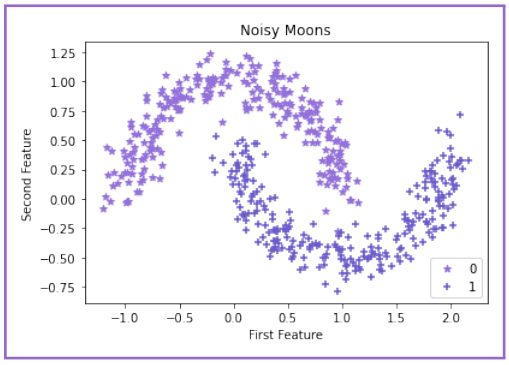

FIGURE 8.1: Noisy Moon Dataset

8.1 Part A

In this part, each class is estimated with one Gaussian distribution. Do to so, prior probabilities are calculated based on the two labels. Secondly, mean and covariance is calcualted based on data in dataset. Usng these values, the likelihood function is evaluated, which its product to prior probabilites, result in posteriors. Now, by employing maximum likelihood estimation, label of each data is evaluated. In addition, with the estimated mean and variance, the contours are computed. The resulting estimation is depicted below:

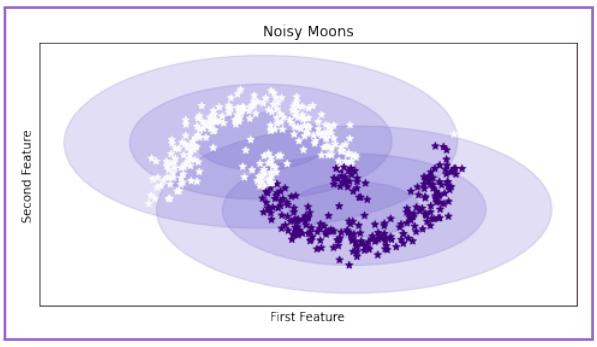

FIGURE 8.2: Noisy Moon Dataset and Its Contours

8.2 Part B

In this part, first of all, the class is implemented which its task it to perform *GMM*. To do so, 3 parameters are required to be set. The number of iterations, the number of components, and the variance smoothing parameter. The last parameter is set, so that while calculating likelihood, in cases where the covariance matrix is singular, we fix it. As the initial step, based on the number of components, mean and covariance values are extracted. Afterward, there are also two prominent parts. One is to implement *E-Step* in which P_j and α_j are updated while μ_j and Σ_j are assumed to be fixed. For updating P_j is updated by likelihood function, while α_j is P_j's mean. The next step is *M-Step* in which μ_j and Σ_j based on the below formulations while P_j and α_j are concidered to be constant.

$$\hat{\mu}_j = \frac{\sum_{i=1}^{n} P_j \underline{x}_i}{\sum_{i=1}^{n} P_j}$$

$$\hat{\Sigma}_j = \frac{\sum_{i=1}^{n} P_j (\underline{x}_i - \hat{\mu}_j)(\underline{x}_i - \hat{\mu}_j)^T}{\sum_{i=1}^{n} P_j}$$

$$P_j = p\{j|\underline{x}_i, \theta^{(g)}\}$$

Then, calculated values of μ_j and Σ_j are employed to create estimated contoures. The resulting plot for various values of components are depicted below:

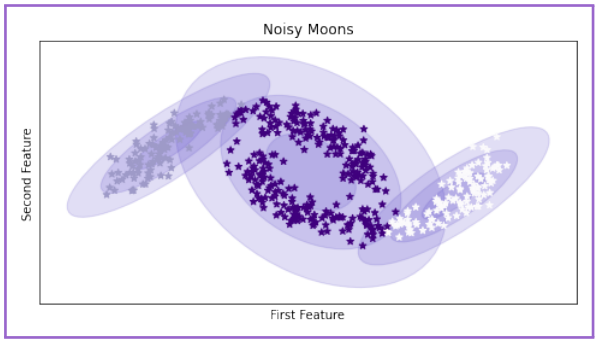

FIGURE 8.3: Noisy Moon Dataset and Its Contours for 3 Components

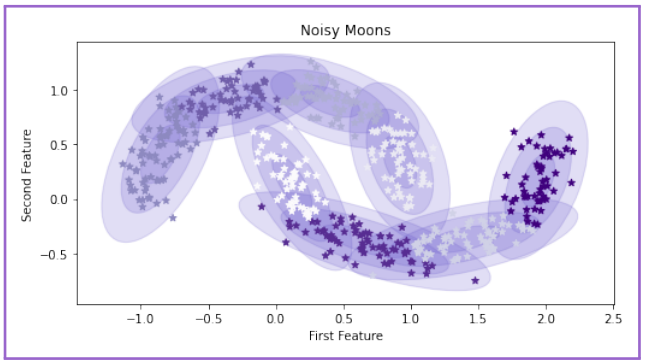

FIGURE 8.4: Noisy Moon Dataset and Its Contours for 8 Components

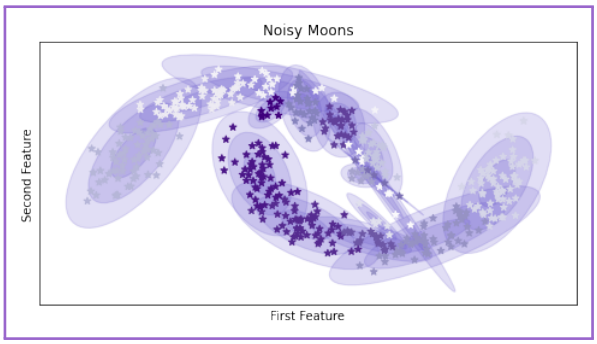

FIGURE 8.5: Noisy Moon Dataset and Its Contours for 16 Components

Afterward, we are asked to plot AIC and BIC for vairous values of 1-16. To do so, AIC and BIC's calculated as below:

$$AIC = 2k - 2ln(L),\ BIC = kln(n) - 2ln(L)$$

where k is the number of independent variables to build the model, L is the maximum likelihood estimate of the model, and n is the sample size (number of observations). Based on the above formilation, the resulting plots will be:

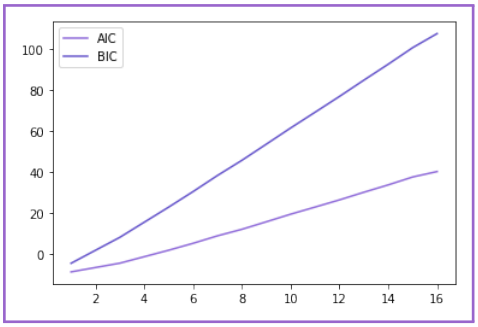

FIGURE 8.6: AIC and BIC of Noisy Moon Dataset

Based on the above formulation as well as the meaning behind AIC and BIC, the best number of components for our dataset is 1. However, if these figure are ploted by the implemented functions is *SKlearn*, the result is:

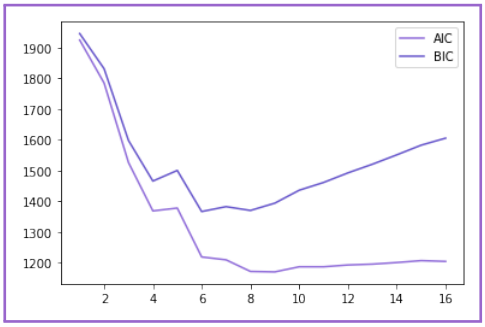

FIGURE 8.7: AIC and BIC of Noisy Moon Dataset

This figure suggests that the best value for the number of components is 8. Although my implementation *GMM* is accurate, since the values of mean, variance, and contours are the same as *SKlearn*'s, the formulation for AIC and BIC are not suitable for this case.