

گزارش تکلیف ۳ درس یادگیری ماشین

کسرا سینایی

شماره دانشجویی ۸۱۰۶۹۶۲۵۴

۲۲ آذر ۱۴۰۰

۱ تمرین آماری

۱.۱ سوال یک

$$\begin{aligned}
 P(\theta | D^1) &= U(0, 10) \xrightarrow{D_1 = \{x_1 = 4\}} P(\theta | D^1) \propto P(x_1 | \theta) P(\theta | D^0) = \begin{cases} \frac{1}{\theta} & \theta \in [4, 10] \\ 0 & \text{ow} \end{cases} \\
 \xrightarrow{D_2 = \{x_1 = 4, x_2 = 7\}} P(\theta | D^2) \propto P(x_2 | \theta) P(\theta | D^1) = \begin{cases} \frac{1}{\theta^2} & \theta \in [7, 10] \\ 0 & \text{ow} \end{cases} \rightarrow \dots \\
 \rightarrow P(\theta | D^n) \propto \begin{cases} \frac{1}{\theta^n} & \max_x [D^n] \leq \theta \leq 10 \\ 0 & \text{ow} \end{cases} \rightarrow P(\theta | D^4) \propto \begin{cases} \frac{1}{\theta^4} & 8 \leq \theta \leq 10 \\ 0 & \text{ow} \end{cases} \\
 \xrightarrow{\text{Max Likelihood}} \hat{\theta} = 8 \quad P(x | D) \sim U(\theta, 8)
 \end{aligned}$$

۲.۱ سوال دو

$$\begin{aligned}
 P(D | \theta) &= \prod_{k=1}^n P(x_k | \theta) = \prod_{k=1}^n \prod_{i=1}^d \theta_i^{x_{ik}} (1 - \theta_i)^{(1-x_{ik})} \xrightarrow{\log} \\
 \ell(\theta) &= \sum_{k=1}^n \sum_{i=1}^d x_{ik} \ln \theta_i + (1 - x_{ik}) \ln (1 - \theta_i) \quad \boxed{\underline{\theta} = \frac{1}{n} \sum_{k=1}^n \underline{x}_k} \\
 \frac{d\ell(\theta)}{d\theta} &= 0 \rightarrow \sum_{k=1}^n \left[\frac{x_{ik}}{\theta_i} + \frac{x_{ik} - 1}{1 - \theta_i} \right] = \sum_{k=1}^n [x_{ik} - \theta_i] = 0 \rightarrow \theta_i = \frac{1}{n} \sum_{k=1}^n x_{ik} \quad \uparrow \text{vector notation}
 \end{aligned}$$

۳.۱ سوال چهار

الف $\mu_n = \left[\frac{1}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}} \right] \frac{n}{\sigma^2} \hat{\mu}_n + \left[\frac{1}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}} \right] \frac{\mu_0}{\sigma_0^2}$ (eliminating σ_n^2 using: $\frac{1}{\sigma_n^2} = \frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}$)

$\xrightarrow{n_0 = \frac{\sigma^2}{\sigma_0^2}}$ $\mu_n = \left[\frac{1}{1 + \frac{\sigma^2}{n\sigma_0^2}} \right] \hat{\mu}_n + \left[\frac{1}{1 + \frac{\sigma^2}{n\sigma_0^2}} \right] \mu_0 = \left[\frac{1}{1 + \frac{\sigma^2}{n\sigma_0^2}} \right] \frac{1}{n} \sum_{k=1}^n x_k + \left[\frac{1}{1 + \frac{\sigma^2}{n\sigma_0^2}} \right] \frac{1}{n_0} \sum_{k=n_0+1}^0 x_k$

$= \frac{1}{n+n_0} \sum_{k=1}^n x_k + \frac{1}{n+n_0} \sum_{k=n_0+1}^0 x_k = \frac{1}{n+n_0} \sum_{k=n_0+1}^n x_k \quad \checkmark$

$\frac{1}{\sigma_n^2} = \frac{n}{\sigma^2} + \frac{1}{\sigma_0^2} \rightarrow \sigma_n^2 = \frac{\sigma^2 \sigma_0^2}{n\sigma_0^2 + \sigma^2} = \frac{\sigma^2}{n + \frac{\sigma^2}{\sigma_0^2}} \xrightarrow{n_0 = \frac{\sigma^2}{\sigma_0^2}} \sigma_n^2 = \frac{\sigma^2}{n+n_0} \quad \checkmark$

ب $\mu_n = \frac{n}{n+n_0} \left[\frac{1}{n} \sum_{k=0}^n n \right] + \frac{n_0}{n+n_0} \left[\frac{1}{n_0} \sum_{k=n_0+1}^0 x_k \right]$ μ_n میانگینی وزن دار داده های واقعی و داده های ساختگی باشد.

* اگر $\frac{1}{\sigma_n^2}$ را معیاری از دقت در نظر بگیریم؛ داریم $\frac{1}{\sigma_n^2} = \frac{n}{\sigma^2} + \frac{n_0}{\sigma^2}$ ، که نشان می دهد دقت همین به صورت خطی با افزایش داده های واقعی بیشتر می شود.

٤.١ سوال هفت

الف

$$\begin{cases} P(a_1), P(a_4) = 0.5 \rightarrow P(a_2) = P(a_3) = 0 \\ P(b_1) = 1 \rightarrow P(b_2) = 0 \\ P(d_2) = 1 \rightarrow P(d_1) = 0 \end{cases}$$

$$P(x_1) = P(x_1|a_1, b_1)P(a_1)P(b_1) + P(x_1|a_4, b_1)P(a_4)P(b_1)$$

$$= \frac{(0.9)(0.65)}{0.9 \times 0.65 + 0.1 \times 0.35} \times 0.5 + \frac{(0.8)(0.65)}{(0.8)(0.65) + (0.2)(0.35)} \times 0.5$$

$$\rightarrow P(x_1) = 0.913 \quad (\text{fish is salmon})^*$$

$$P(x_2) = P(x_2|a_1, b_1)P(a_1)P(b_1) + P(x_2|a_4, b_1)P(a_4)P(b_1) = \frac{0.35 \times 0.1}{(0.35)(0.1) + (0.9)(0.65)} \times 0.5 + \frac{0.2 \times 0.35}{0.8 \times 0.65 + 0.2 \times 0.35} \times 0.5$$

$$\rightarrow P(x_2) = 0.087 \quad (\text{sea bass})^*$$

evidence is only based on wide/thin:

$$P(x_1) = P(e|d_1)P(d_1) + P(e|d_2)P(d_2) = 0 + 1 \times 0.6 = 0.6^{**}$$

Normalize $^*, ** \rightarrow P(x_1|e) = \frac{0.913 \times 0.6}{0.913 \times 0.6 + 0.087 \times 0.4} = 0.992$

$$P(x_2|e) = \frac{0.087 \times 0.4}{0.913 \times 0.6 + 0.087 \times 0.4} = 0.008$$

fish is salmon
P-error = 0.008

$$\begin{aligned} \therefore & \begin{cases} P(e_D | d_1) = 0 \rightarrow P(e_D | d_2) = 1 \\ P(e_C | c_1) = 0 \rightarrow P(e_C | c_2) = 1 \end{cases} \\ P(x_1) &= P(e_C | x_1) P(e_D | x_1) = [P(e_C | c_1) P(c_1 | x_1) + P(e_C | c_2) P(c_2 | x_1) + P(e_C | c_3) P(c_3 | x_1)] \\ & \quad \times [P(e_D | d_1) P(d_1 | x_1) + P(e_D | d_2) P(d_2 | x_1)] \\ \rightarrow P(x_1) &= (0 + 1 \times 0.33 + 0)(0 + 1 \times 0.6) = 0.198 \\ P(x_2) &= (0 + 1 \times 0.1 + 0)(0 + 1 \times 0.05) = 0.005 \\ P(x_1 | e) &= \frac{0.198}{0.198 + 0.005} \approx 0.975, \quad P(x_2 | e) = \frac{0.005}{0.198 + 0.005} \approx 0.025 \Rightarrow \boxed{\text{Salmon } (x_1)} \\ P(a_1) + P(a_2) + P(a_3) + P(a_4) &= \frac{1}{4} \xrightarrow{\text{Salmon}} P(a_i | x_1) = \frac{P(x_1 | a_i) P(a_i)}{\sum_{i=1}^4 P(x_1 | a_i) P(a_i)} \rightarrow \begin{cases} P(a_1) = 0.375 \\ P(a_2) = 0.125 \\ P(a_3) = 0.167 \\ P(a_4) = 0.333 \end{cases} \\ & \quad \text{(Winter) } a_1 \end{aligned}$$

$$\begin{aligned} \therefore & \begin{cases} P(x_1 | e) = 0.975 \\ P(x_2 | e) = 0.025 \end{cases} \rightarrow \text{Salmon } (x_1) \\ \text{North Atlantic} & \rightarrow P(b_1 | e_B) = 1, P(b_2 | e_B) = 0 \\ \Rightarrow & \begin{cases} P(a_1 | x_1, b_1) = P(x_1 | a_1) P(x_1 | b_1) P(a_1) P(b_1) = 0.9 \times 0.65 \times 0.25 = 0.146 \\ P(a_2 | x_1, b_1) = 0.3 \times 0.65 \times 0.25 = 0.049 \\ P(a_3 | x_1, b_1) = 0.4 \times 0.65 \times 0.25 = 0.065 \\ P(a_4 | x_1, b_1) = 0.8 \times 0.65 \times 0.25 = 0.13 \end{cases} \\ \Rightarrow & \text{Winter}, \quad P(x_1 | e) P(a_1 | x_1) = 0.975 \times 0.146 = \boxed{0.367} \end{aligned}$$

۲ تمرین شبیه سازی

۱.۲ سوال هشت

در این تمرین ابتدا با استفاده از کتابخانه OpenCV عکس‌های موجود در پوشه مربوط به این سوال را لود کرده و تنها مقادیر مربوط به میانگین کانال R و B را به عنوان فیچر استخراج شده و به همراه لیبِل‌ها باز گردانده می‌شوند. تابع مورد استفاده در این سوال مشابه تمرین یک بوده و در فایل load_pics.py پیاده سازی شده است. برای پیاده سازی طبقه بند در هر سه سوال این تکلیف یک کلاس به نام GMM پیاده سازی شده است که سه متد برای فیت کردن مدل و پیش‌بینی دارد:

- **constructor**: ابعاد فیچرها، تعداد کلاس‌ها و تعداد المان‌های میکسچر مدل را از کاربر می‌گیرد و یک لیست آبجکت mixture Gaussian (از کتابخانه scikit learn) با تعداد میکسچر تنظیم شده در آرگومان ورودی می‌سازد.

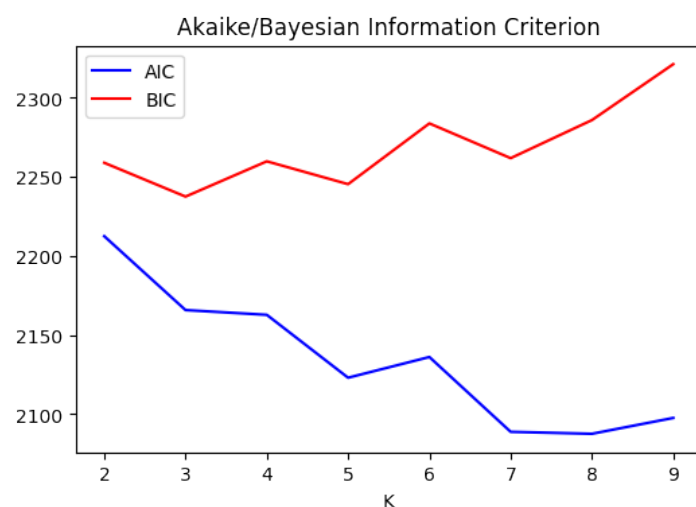
- **update**: داده‌ها و لیبِل‌ها را گرفته و برای هر کلاس پارامترهای میکسچر مدل را آپدیت می‌کند. (با استفاده از متد fit)

- **predict**: این تابع یک بردار با اندازه دلخواه از فیچرها می‌گیرد و با توجه به میکسچر مدل‌های یادگرفته شده لیبِل آن‌ها را پیش‌بینی می‌کند. مبنای تصمیم‌گیری در این طبقه بند loglikelihood میکسچر مدل‌ها برای هر یک از کلاس‌ها است. با استفاده از متد score samples این مقدار به دست می‌آید.

تابع update حاصل جمع مقادیر AIC و BIC را بازمی‌گرداند. میانگین‌ها و ماتریس کواریانس همه کلاس‌ها نیز پابلیک می‌باشد و می‌توان خراج از کلاس به آن‌ها دسترسی داشت و از آن‌ها برای ترسیم کانتورهای میکسچر مدل‌های کلاس‌ها استفاده کرد.

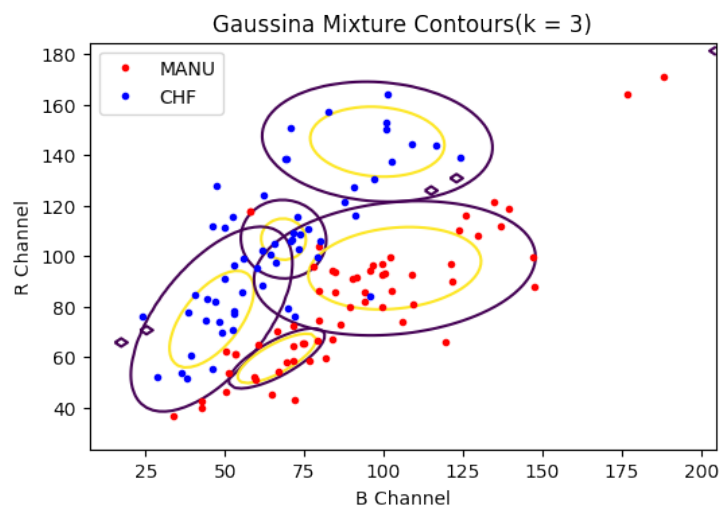
تابع plot برای ترسیم این کانتورها در هر سه سوال شبیه سازی این تکلیف استفاده شده است. این تابع چهار آرایه numpy در آرگومان‌های ورودی خود دریافت می‌کند. آرایه اول در هر سطر خود میانگین یک توزیع گوسی را دارد، آرایه دوم ماتریس کواریانس متناظر با میانگین توزیع‌های آرایه اول است، آرایه سوم و چهارم داده‌های سوال هستند که به صورت نقاط رنگی اسکتر می‌شوند. برای رسم کانتور نیز یک مش‌گرید در راستای x و y تشکیل می‌دهیم و با استفاده از تابع Multivariate Normal از کتابخانه scipy احتمال آن را حساب می‌کنیم و برای هر توزیع نرمال دو خط کانتور رسم می‌کنیم.

نمودارهای AIC و BIC را پس از فیت کردن میکسچر مدل‌هایی با تعداد مدل ۲-۹ رسم می‌کنیم تا بر اساس آن بهینه‌ترین مدل برای این سوال را بیابیم.



شکل ۱: نمودارهای Akaike/Bayes Information Criterion به ازای مقادیر مختلف k

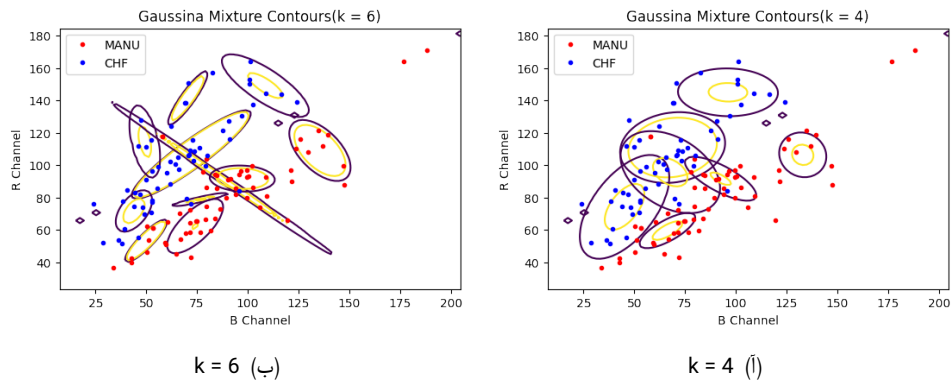
با توجه به نمودار می‌توان نتیجه گرفت مقدار ۳ و ۷ برای مدل مناسب به نظر می‌رسند. حال کانتورهای مربوط به میکسچر مدل را برای این طبقه بند رسم می‌کنیم.



شکل ۲: کانتور توزیع‌های گاوسی مربوط به میکسچر مدل‌های دو کلاس

برای این حالت پارامترهای مدل‌های گاوسی فیت شده برای هر کلاس در جدول زیر گزارش شده‌اند. همچنین طبقه‌بند با تعداد مؤلفه‌های بالاتر (۴ و ۶) نیز آپدیت شد و کانتور مدهای گاوسی آن پس از یادگیری در شکل زیر نشان داده شده‌اند. با توجه به مقدار BIC تعداد مؤلفه بهینه برای این مسئله ۳ انتخاب شد تأثیر افزایش مؤلفه‌ها بر روی شکل

کانتور توزیع‌های گاوسی طبقه بند برای هر یک از کلاس‌ها در شکل دیده می‌شود. تطابق بیش از اندازه بر روی داده‌های نویزی (عکس‌هایی که هم رنگ آبی و هم رنگ قرمز در آن‌ها به یک مقدار دیده می‌شد) لزوماً به بهبود عملکرد طبقه بند نمی‌انجامد.



شکل ۳: کانتورهای به دست آمده از طبقه‌بند با تعداد مؤلفه‌های بیشتر

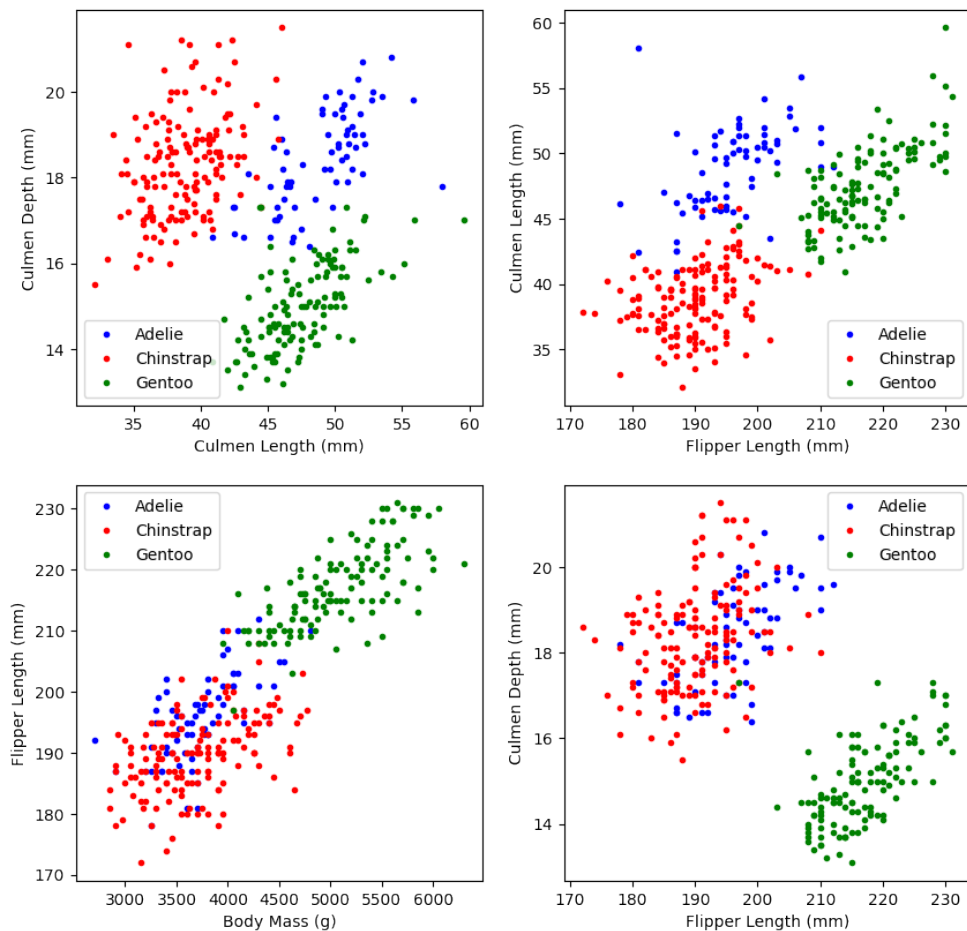
CHF			ManU			comp
σ	μ	weight	σ	μ	weight	
$\begin{pmatrix} 33.6 & 20.7 \\ 20.7 & 12.8 \end{pmatrix}$	$(182.4 \quad 167.4)$	0.03	$\begin{pmatrix} 169.1 & -4.9 \\ -4.9 & 226.12 \end{pmatrix}$	$(68.4 \quad 106.6)$	0.46	1
$\begin{pmatrix} 232.9 & 138.5 \\ 138.5 & 144.8 \end{pmatrix}$	$(66.3 \quad 59.4)$	0.44	$\begin{pmatrix} 99.9 & 87.3 \\ 87.3 & 207.4 \end{pmatrix}$	$(45.8 \quad 74.9)$	0.34	2
$\begin{pmatrix} 536.8 & 54.6 \\ 54.6 & 193.0 \end{pmatrix}$	$(103.5 \quad 95.1)$	0.53	$\begin{pmatrix} 250.2 & -15.8 \\ -15.8 & 107.1 \end{pmatrix}$	$(98.0 \quad 145.2)$	0.19	3

جدول ۱: پارامترهای مدل فیت شده

۲.۲ سوال نه

در این سوال ابتدا داده‌های سه کلاس را بر حسب فیچرهای خواسته شده دو به دو در نمودار به تصویر می‌کشیم تا تشخیص بدهیم کدام فیچرها برای طبقه‌بندی مناسب‌تر می‌باشند.

Feature Visualization



شکل ۴: نمونه‌های مورد استفاده برای طبقه‌بندی پنگوئن‌ها

تفکیک‌پذیری داده‌ها بر حسب دو فیچر culmen length و culmen depth از سایر مجموعه فیچرهای خواسته شده در صورت سوال بهتر به نظر می‌رسند. داده‌های مربوط به سه گونه مختلف پنگوئن بر حسب این دو ویژگی از یکدیگر بیشتر فاصله داشته و نمونه‌های استثنایی کمتر به چشم می‌خورند.

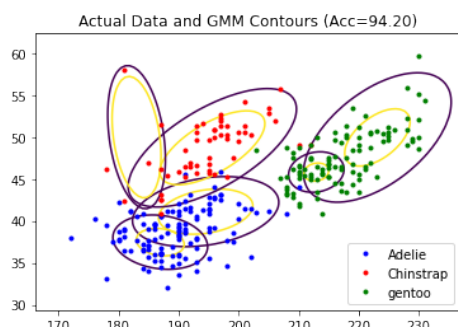
در قسمت بعدی تمرین با استفاده از کلاس GMM که در سوال ۸ همین تمرین نیز از آن استفاده شد، طبقه‌بندیهایی بر مبنای میکسچر مدل‌های گوسی با تعداد مؤلفه‌های مختلف فیت می‌کنیم. بر حسب دقت طبقه‌بند بر روی مجموعه

داده‌ی تست مشخص می‌کنیم که کدام مجموعه فیچر برای طبقه بندی بهتر می‌باشد. دقت حالات مختلف طبقه بندی در جدول زیر آورده شده‌اند.

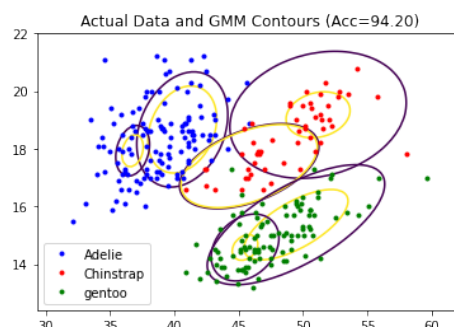
GMM Components	Feature Set	Accuracy
۲	۱	98.5
	۲	95.6
	۳	79.7
	۴	78.2
۳	۱	97.1
	۲	95.6
	۳	81.1
	۴	84.0
۴	۱	97.1
	۲	95.6
	۳	81.1
	۴	81.1
۵	۱	98.5
	۲	95.6
	۳	81.1
	۴	81.1

جدول ۲: دقت مدل‌های فیت بر روی مجموعه فیچرهای مختلف

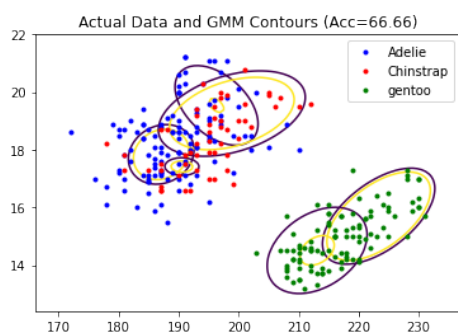
در داده‌های این سوال تعدادی از فیچرها در برخی از سمپل‌ها ناقص و نامعلوم بودند. با توجه به اینکه تعداد این موارد در کل دیتاست درصد قابل اغمازی بود با استفاده از imputer میانه، فیچرهای نامعلوم جاگذاری شدند. انتظار می‌رود تغییر چندانی در عملکرد نهایی طبقه بند مشاهده نشود، چون تعداد این موارد جمعاً به ده نمی‌رسید. برای رسم کانتور توزیع‌های فیت شده با استفاده از زیرمجموعه‌های مختلف از دیتاست تعداد مؤلفه‌ها را برابر با ۲ در کلاس GMM تنظیم می‌کنیم و با دیتاست‌های مختلف مدل را آپدیت می‌کنیم. کانتورهای به دست آمده برای برای هر زیرمجموعه از دیتاست و دقت طبقه بند بر روی مجموعه تست در تصاویر زیر نشان داده شده‌اند.



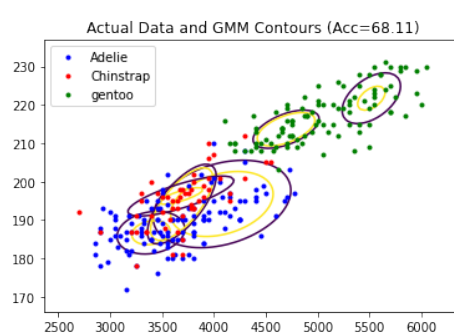
(ب) فیچرهای ۲



(ا) فیچرهای ۱



(د) فیچرهای ۴

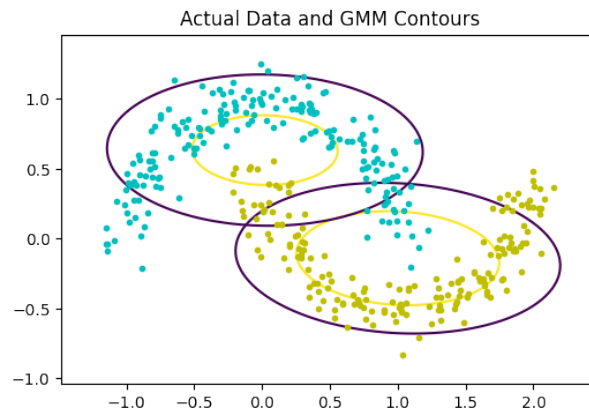


(ج) فیچرهای ۳

شکل ۵: کانتور توزیع‌های گاوسی مربوط به میکسچرمدل با تعداد مؤلفه‌های متفاوت

۳.۲ سوال ده

در این سوال عملکرد طبقه‌بند ساده بیزی را با عملکرد طبقه‌بندی با مدل GMM مقایسه می‌کنیم. ابتدا همانند تکالیف قبلی با استفاده از الگوریتم optimal bayes و بدون دانش اولیه (prior) داده‌ها را طبقه‌بندی می‌کنیم. کانتور مدل‌های گاوسی به دست آمده برای تصمیم‌گیری بیز در شکل زیر نشان داده شده‌اند. دقت طبقه‌بند در این حالت برابر ۸۸ درصد می‌باشد.



شکل ۶: توزیع‌های به دست آمده توسط تخمین‌گر بیزی ساده

برای آنکه بتوان از کلاس GMM نوشته شده برای سؤالات قبلی استفاده کرد یک کلاس به نام GaussianMixture همانند آبجکت موجود در کلاس سای‌کیت لرن پیاده‌سازی می‌کنیم. نام متغیرهای غیرخصوصی این کلاس مانند means, covariances و weights می‌گذاریم. متدهای زیر برای اجرای الگوریتم EM باید پیاده‌سازی شوند.

- constructor: آرایه‌های نام‌پای با طول متناسب با ابعاد فیچر و تعداد مؤلفه‌های میکسچر مدل برای میانگین، واریانس و وزن توزیع‌ها تشکیل می‌دهد.
- fit: داده‌های ترین را به همراه تعداد ایتريشن می‌گیرد و به طور متوالی E Step و M Step را انجام می‌دهد.
- e step: این متد تابع Q را برای متغیر مشاهده نشده فعلی Z به روز رسانی می‌کند. از این متغیر در قسمت دوم الگوریتم استفاده خواهد شد.
- m step: این متد پارامترهای میکسچر مدل را با بیشینه‌سازی تابع Q به دست آمده از قسمت الگوریتم به روزرسانی می‌کند.
- log likelihood: لوگ‌لایکلیت را برای داده‌های آرگومان ورودی بر حسب پارامترهای فعلی مدل محاسبه می‌کند. این تابع برای محاسبه AIC و BIC لازم می‌شود.
- aic: مقدار AIC را با استفاده از رابطه

$$AIC(G) = -2 \log \ell_o(\hat{\theta}_G; G) + 2v_G$$

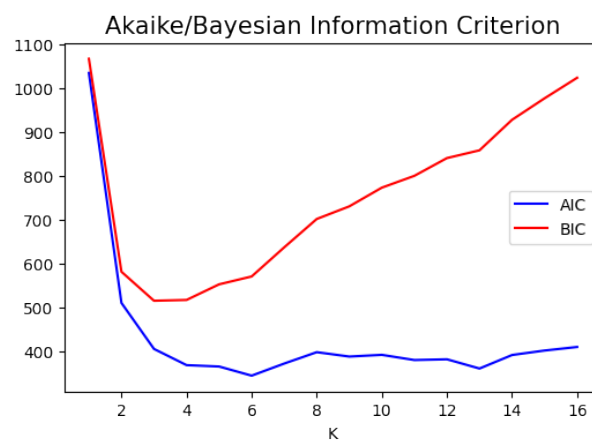
محاسبه می‌کند

• bic: مقدار BIC را با استفاده از رابطه

$$\text{BIC}(G) = -2 \log \ell_o(\hat{\theta}_G; G) + v_G \log(n)$$

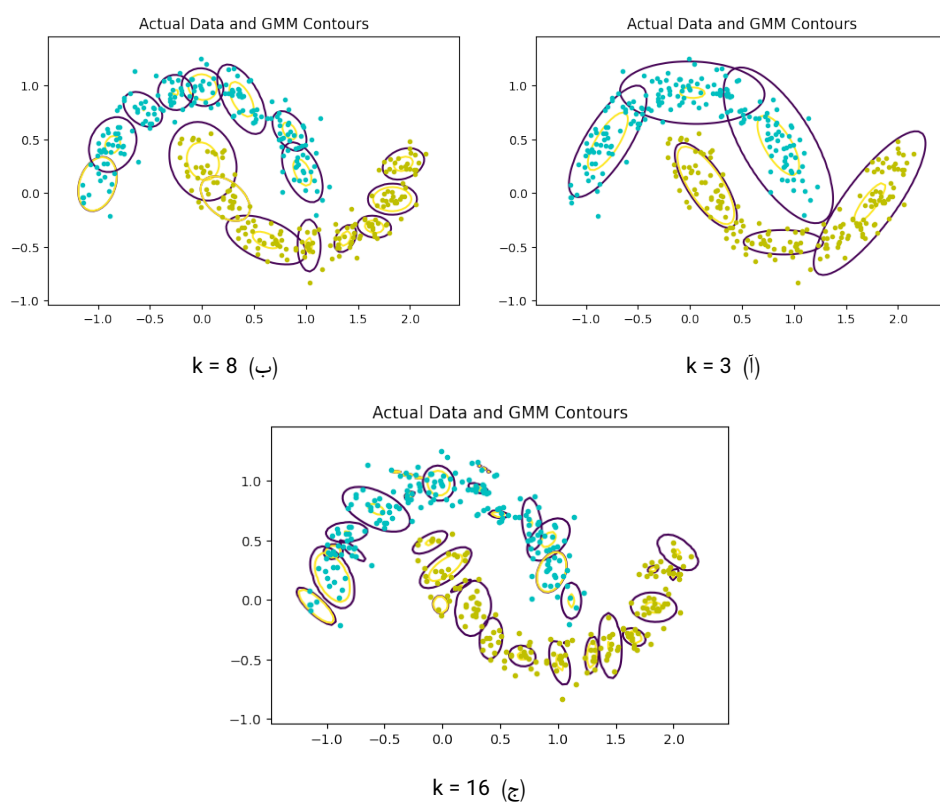
محاسبه می‌کند.

حال برای تشخیص دادن بهترین میکسچر مدل برای طبقه‌بند این سوال تعداد مؤلفه‌های میکسچر مدل طبقه‌بند را از ۲ تا ۱۶ تغییر می‌دهیم و با رسم نمودار AIC و BIC بهینه‌ترین تعداد برای مؤلفه‌ها را می‌یابیم. این اطلاعات در شکل زیر نمایش داده شده‌اند.



شکل ۷: تاثیر تعداد مؤلفه‌های میکسچر مدل گاوسی

برای رسم نمودارها و طبقه‌بندی در این سوال نیز همان کلاس پیاده سازی شده در سوال ۸ استفاده شده است.



شکل ۸: کانتور توزیع‌های گاوسی مربوط به میکسچر مدل با تعداد مؤلفه‌های متفاوت