به نام خدا

دانشگاه تهران

دانشکدگان فنی

دانشکده مهندسی برق و کامپیوتر

یادگیری ماشین

تمرین پنجم

نام و نام خانوادگی : حسین سیفی

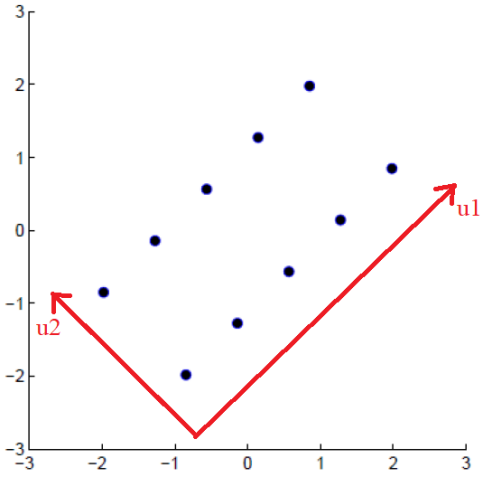
شماره دانشجویی : ۸۱۰۱۰۰۳۸۶

تیر ۱۴۰۲

سوال ۱

الف

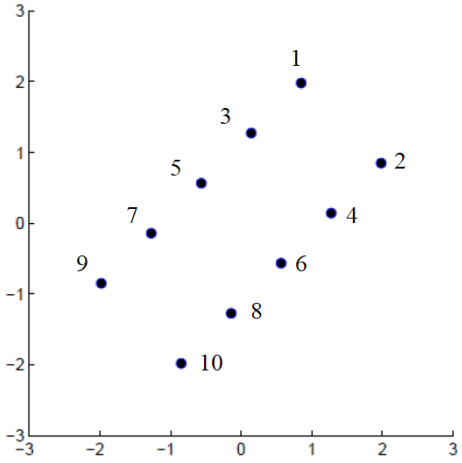
محورهای اول و دوم PCA به ترتیب به صورت u1 و u2 در شکل زیر به نمایش درآمده‌اند:



روش کاهش بعد PCA یک روش Unsupervised است و برای انتخاب محورهای جدید نیازی به برچسب‌های دادگان وجود ندارد و تنها تلاش می‌کند تا محوری را انتخاب کند که با نگاشت داده‌ها بر روی محور جدید، بیشترین مقدار واریانس حاصل شود.

ب

داده‌های نمایش داده شده در نمودار فوق را به شکل زیر شماره گذاری می‌کنیم:



سپس برای هر کدام از شرایط گفته شده داده‌ها را به شکل زیر برچسب می‌زنیم:

ب.۱

Point Num.	1	2	3	4	5	6	7	8	9	10
Label	+	+	-	-	+	+	-	-	+	+

با توجه به انتساب برچسب‌های فوق، برای مثال در صورتی که نقطه ۳ به عنوان نمونه تست انتخاب شود، در دو بعد نزدیک‌ترین نمونه به نقطه ۳ نقاط ۱ و ۵ هستند و هر دو برچسب + دارند و مدل نزدیک‌ترین همسایه برچسب + را برای نقطه ۳ انتخاب می‌کند در حالی که برچسب اصلی

– است. در حالتی که از یک بعد PCA استفاده کنیم، نزدیک‌ترین همسایه به نقطه ۳، نمونه ۴ خواهد بود که برچسب – دارد و در نتیجه برچسب صحیح برای نقطه ۳ انتخاب خواهد شد. در صورت انتخاب سایر نقطه‌ها به عنوان نمونه تست نیز شرایط به شکل توصیف شده خواهد بود و در نتیجه خطا در دو بعد برابر با ۱۰۰ درصد و در یک بعد PCA برابر با ۰ درصد خواهد بود.

ب.۲

Point Num.	1	2	3	4	5	6	7	8	9	10
Label	+	-	+	-	+	-	+	-	+	-

در صورتی که برچسب‌های فوق را به عنوان برچسب دادگان انتخاب کنیم، و نقطه ۳ به عنوان نمونه تست انتخاب شود، در دو بعد نزدیک‌ترین نقاط، نمونه‌های ۱ و ۵ هستند و هر دو برچسب + دارند و مدل نزدیک‌ترین همسایه برچسب صحیح را برای این نمونه انتخاب می‌کند. در حالی که در یک بعد PCA نزدیک‌ترین نقطه، نقطه‌ی ۴ خواهد بود که برچسب – دارد و مدل نزدیک‌ترین همسایه برچسب اشتباه را انتخاب می‌کند. در صورت انتخاب سایر نقطه‌ها به عنوان نمونه تست نیز شرایط به شکل توصیف شده خواهد بود و در نتیجه خطا در دو بعد برابر با ۰ درصد و در یک بعد PCA برابر با ۱۰۰ درصد خواهد بود.

سوال ۲

برای رسیدن به عبارت نهایی از عبارت ابتدایی، میانگین هر دو دسته را به شکل زیر اضافه و کم می‌کنیم:

$$J = \frac{1}{n_1 n_2} \sum_{y_i \in Y_1} \sum_{y_j \in Y_2} \Big((y_i - m_1) - (y_j - m_2) + (m_1 - m_2) \Big)^2$$

سپس با اعمال توان ۲ بر عبارت داخل پرانتز، آن را باز می‌کنیم:

$$J = \frac{1}{n_1 n_2} \sum_{y_i \in Y_1} \sum_{y_j \in Y_2} [(y_i - m_1)^2 + (y_j - m_2)^2 + (m_1 - m_2)^2 - 2(y_i - m_1)(y_j - m_2) + 2(y_i - m_1)(m_1 - m_2) - 2(y_j - m_2)(m_1 - m_2)]$$

عبارت فوق را با استفاده از جمع و تفریق‌های موجود تفکیک می‌کنیم و عباراتی که به هر یک از Σها وابسته نیستند از آن خارج می‌کنیم:

$$J = \frac{n_2}{n_1 n_2} \sum_{y_i \in Y_1} (y_i - m_1)^2 + \frac{n_1}{n_1 n_2} \sum_{y_j \in Y_2} (y_j - m_2)^2 + \frac{n_1 n_2}{n_1 n_2} (m_1 - m_2)^2 - \frac{2}{n_1 n_2} \sum_{y_i \in Y_1} \sum_{y_j \in Y_2} (y_i - m_1)(y_j - m_2) + \frac{2n_2}{n_1 n_2} \sum_{y_i \in Y_1} (y_i - m_1)(m_1 - m_2) - \frac{2n_1}{n_1 n_2} \sum_{y_j \in Y_2} (y_j - m_2)(m_1 - m_2)$$

اثبات می‌شود که سه عبارت خط دوم برابر با صفر هستند(هر سه به دلیل محاسبه مجموعه بر روی تفریق میانگین از هر نمونه) و بنابراین از عبارت فوق حذف می‌شوند. عبارت پس از ساده‌سازی به شکل زیر قابل بازنویسی است:

$$J = \frac{1}{n_1} \sum_{y_i \in Y_1} (y_i - m_1)^2 + \frac{1}{n_2} \sum_{y_j \in Y_2} (y_j - m_2)^2 + (m_1 - m_2)^2$$

طبق تعریف پراکندگی نمونه‌ها، جملات اول و دوم در عبارت فوق به ترتیب پراکندگی نمونه‌های دسته‌های ۱ و ۲ را نشان می‌دهد:

$$J = \frac{1}{n_1} S_1^2 + \frac{1}{n_2} S_2^2 + (m_1 - m_2)^2$$

و در نهایت اثبات شد که عبارت ابتدایی را می‌توان به شکل ثانویه بازنویسی کرد.

اثبات صفر شدن یکی از عبارات مطرح شده:

$$\frac{1}{n_2} \sum_{y_j \in Y_2} (y_j - m_2)(m_1 - m_2) = (m_1 - m_2)[\frac{1}{n_2} \sum_{y_j \in Y_2} y_j - \frac{1}{n_2} \sum_{y_j \in Y_2} m_2] = (m_1 - m_2) \left[m_2 - \frac{n_2}{n_2} m_2 \right] = 0$$

سوال ۳

الف

در این سوال، محاسبات الگوریتم EM با فرض اینکه داده‌ها از توزیع پوآسون باشند، برای ۲ کامپوننت انجام می‌شود. توزیع کلی داده‌های موجود با کمک عبارت زیر که یک ترکیب خطی با ضریب α از توزیع دو کامپوننت داده‌ها است، به دست می‌آید:

$$P(x|\lambda) = (1-\alpha)Poisson(x;\lambda_1) + \alpha Poisson(x;\lambda_2)$$

با در نظر گرفتن متغیر نهان(y) که دارای مقدار ۰ یا ۱ است که کامپوننت هر داده را مشخص می‌کند، عبارت فوق را بازنویسی می‌کنیم:

$$P(x,y|\lambda) = [(1-\alpha)Poisson(x;\lambda_1)]^{1-y}[\alpha Poisson(x;\lambda_2)]^{y}$$

گام E:

حال مرحله E را با در نظر گرفتن فرض ساده‌ساز استقلال ویژگی‌ها[1] و با گرفتن Expectation نسبت به متغیر نهان شروع می‌کنیم:

$$E_y[L(\lambda)] = E_y[\sum_{i=1}^{n} logP(x_i,y_i|\lambda)] = \sum_{i=1}^{n} E_y[logP(x_i,y_i|\lambda)]$$

سپس عبارت به دست آمده در مرحله قبل را در عبارت فوق جایگذاری می‌کنیم:

$$E_y[L(\lambda)] = \sum_{i=1}^{n} E_y[(1-y_i)\log(1-\alpha) + (1-y_i)logPoisson(x_i;\lambda_1) + y_i\log\alpha + y_i logPoisson(x_i;\lambda_2)]$$

با توجه به اینکه Expectation بر روی متغیر y محاسبه می‌شود، می‌توان عباراتی که شامل y نیستند را از E[] خارج کرد:

$$E_y[L(\lambda)] = \sum_{i=1}^{n} [\log(1-\alpha)(1-E[y_i]) + logPoisson(x_i;\lambda_1)(1-E[y_i]) + E[y_i]\log\alpha + E[y_i]logPoisson(x_i;\lambda_2)]$$

همچنین می‌توان مقدار E[y] را به شکل زیر محاسبه کرد:

$$E[y] = E[y|x] = P(y=1|x) = \frac{P(x|y=1)P(y=1)}{P(x)}$$

با جایگذاری عبارات متناظر با هر کدام از احتمالات فوق به عبارت زیر می‌رسیم:

$$E[y] = \frac{Poisson(x|\lambda_2^t)\alpha^t}{(1-\alpha^t)Poisson(x;\lambda_1^t) + \alpha^t Poisson(x;\lambda_2^t)} = \gamma^{t+1}$$

سپس E[y_i] را با γ_i جایگزین می‌کنیم:

[1] i.i.d

$$E_y[L(\lambda)] = \sum_{i=1}^{n}[\log(1-\alpha)(1-\gamma_i^t) + logPoisson(x_i;\lambda_1)(1-\gamma_i^t) + \gamma_i^t \log\alpha + \gamma_i^t logPoisson(x_i;\lambda_2)]$$

با به دست آوردن عبارت فوق، مرحله E به پایان می‌رسد. سپس به انجام محاسبات مرحله M می‌پردازیم. در محاسبات مرحله بعد، عبارت فوق را $Q(\lambda)$ می‌نامیم.

گام M:

در این مرحله از $Q(\lambda)$ نسبت به پارامترهای موجود(λ و α) مشتق می‌گیریم و برابر با صفر قرار می‌دهیم تا مقدار این پارامترها مشخص شود:

$$\frac{\partial Q}{\partial \alpha} = \sum_{i=1}^{n}\left[\frac{-1}{1-\alpha}(1-\gamma_i^t) + \frac{\gamma_i^t}{\alpha}\right] = 0 \rightarrow \alpha = \frac{\sum_{i=1}^{n}\gamma_i^t}{n}$$

$$\frac{\partial Q}{\partial \lambda} = 0$$

ب

می‌دانیم که تصویر بردار X بر روی بردار ویژه‌ی اول(u_1) به شکل ضرب داخلی آن‌ها نوشته می‌شود:

$$projection(X) = X^T u_1$$

می‌توان نحوه محاسبه واریانس داده‌های X را به صورت برداری را به شکل زیر نشان داد:

$$\sigma^2 = \frac{1}{n-1}X^T X$$

حال می‌توان واریانس تصویر بردار X بر روی بردار ویژه‌ی اول را به شکل زیر نوشت:

$$\sigma^2 = \frac{1}{n-1}(X^T u_1)^T (X^T u_1)$$

در ادامه می‌توان عبارت فوق را به شکل زیر بازنویسی کرد:

$$\sigma^2 = \frac{1}{n-1}u_1^T X X^T u_1$$

با توجه به تعریف $C = XX^T$ می‌توان جایگذاری زیر را انجام داد:

$$\sigma^2 = \frac{1}{n-1}u_1^T C u_1$$

همچنین می‌دانیم که با توجه به تعریف مسئله eigenvalue-eigenvector عبارت $Cu_1 = \lambda_1 u_1$ صحیح است و می‌توان جایگذاری زیر را انجام داد:

$$\sigma^2 = \frac{1}{n-1}u_1^T \lambda_1 u_1$$

می‌دانیم که λ_1 یک اسکالر است و قابلیت جابجایی پذیری در یک ضرب ماتریسی را دارد و با ساده‌سازی می‌توان نرم ۲ بردار ویژه اول را در عبارت ظاهر کرد:

$$\sigma^2 = \frac{\lambda_1}{n-1} u_1^T u_1 = \frac{\lambda_1}{n-1} ||u_1||^2$$

همچنین با توجه به تعریف PCA می‌دانیم که طول بردارهای ویژه به دست آمده برابر ۱ است و می‌توان $||u_1||^2$ را در عبارت فوق در نظر نگرفت. بنابراین اثبات می‌شود که مقدار واریانس تصویر داده‌ها بر روی بردار ویژه اول برابر با تابعی خطی از مقدار ویژه‌ی اول است.

$$\sigma^2 = \frac{\lambda_1}{n-1}$$

با تعریف ماتریس کوواریانس داده‌ها به شکل $C = \frac{1}{n-1} XX^T$ می‌توان مقدار واریانس تصویر داده‌ها را ساده کرد و مقدار واریانس دقیقا برابر با مقدار ویژه‌ی اول می‌شود.

$$if\ C = \frac{1}{n-1} XX^T\ then\ \sigma^2 = \lambda_1$$

سوال ۴

الف

نتیجه "الف" مناسب‌تر است، چون:

۱. تعداد داده‌ها بیشتری از هر کلاس را نسبت به نتیجه "ب" تحت پوشش قرار داده است.
۲. قسمت کمتری از نواحی که هیچ نقطه‌ای از خوشه مورد نظر قرار نگرفته است نسبت به نتیجه "ب" احتمال تخمین زده است.

ب

به نظر می‌رسد نتیجه اولین گام EM و استفاده از GMM[۲] با توجه به اطلاعات داده شده شکل "الف" باشد چرا که الگوریتم EM با محاسبه میانگین و ماتریس کوواریانس تمامی داده‌های موجود بر اساس توزیع‌های Fit شده ابتدایی(چهار دایره در شکل اول)، به مرور به پوشش حداکثری داده‌ها بپردازد. همچنین از الگوریتم EM انتظار می‌رود با توجه به استفاده از ماتریس کوواریانس داده‌ها(بر خلاف روش‌های خوشه‌بندی مانند K-means که از کوواریانس استفاده نمی‌کنند) با تغییر شکل منطقه دارای بیشتری احتمال، بیشترین احتمال را برای نواحی شامل نقاط تخمین بزند و مناطقی که هیچ نقطه‌ای در آن‌ها قرار نمی‌گیرد دارای احتمال کمتری باشند به همین دلیل در شکلی "الف" که به نظر می‌رسد نتیجه مناسب‌تری برای الگوریتم EM باشد، نواحی مشخص شده به شکل کشیده هستند و تنها به نواحی که شامل توده متراکمی از داده‌ها است احتمال بیشتری تخصیص یابد.

[2] Gaussian Mixture Model

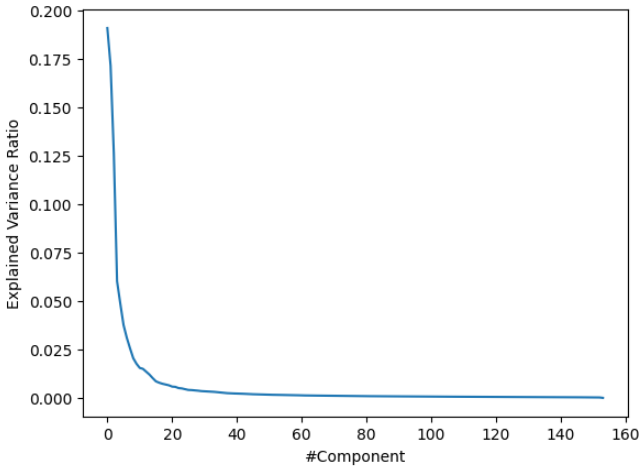
سوال ۵

بخش اول

در این سوال ابتدا به پیاده‌سازی PCA با استفاده از کتابخانه Sklearn می‌پردازیم.

الف

الگوریتم PCA با استفاده از بخش train مجموعه داده مورد نظر پیاده‌سازی شد. در نمودار زیر مقدار ویژه‌ها به ترتیب نزولی بر اساس Explained Variance Ratio مرتب شده و به نمایش درآمده‌اند:



در سوالات تئوری اثبات شد که مقدار ویژه برابر با واریانس تصویر بر بردار ویژه است بنابراین مقادیر نشان داده شده در نمودار فوق به عنوان Explained Variance Ratio همان مقادیر ویژه هستند.

راه‌های زیادی برای تعیین بهترین تعداد کامپونت‌ها وجود دارد که در ادامه معرفی شده‌اند:

۱. استفاده از نمودار تجمعی Explained variance: می‌توان نمودار رسم شده فوق را به صورت تجمعی بر اساس تعداد کامپونت‌ها رسم کرد. با بررسی نمودار ترسیم شده تعداد کامپونتی را انتخاب می‌کنیم که شیب نمودار در آن نقطه به آرامی در حال نزدیک شدن به صفر است.
۲. Cross Validation: می‌توان با استفاده از Cross Validation تعداد کامپونت‌های متفاوت را پیاده‌سازی کرد و مدلی با بهترین عملکرد را تشخیص داد. سپس تعداد کامپونتی که باعث بهترین عملکرد مدل‌ها شده را انتخاب کرد.
۳. استفاده از معیارهای AIC و BIC

ب

۴ مقدار ویژه اول به شرح زیر هستند:

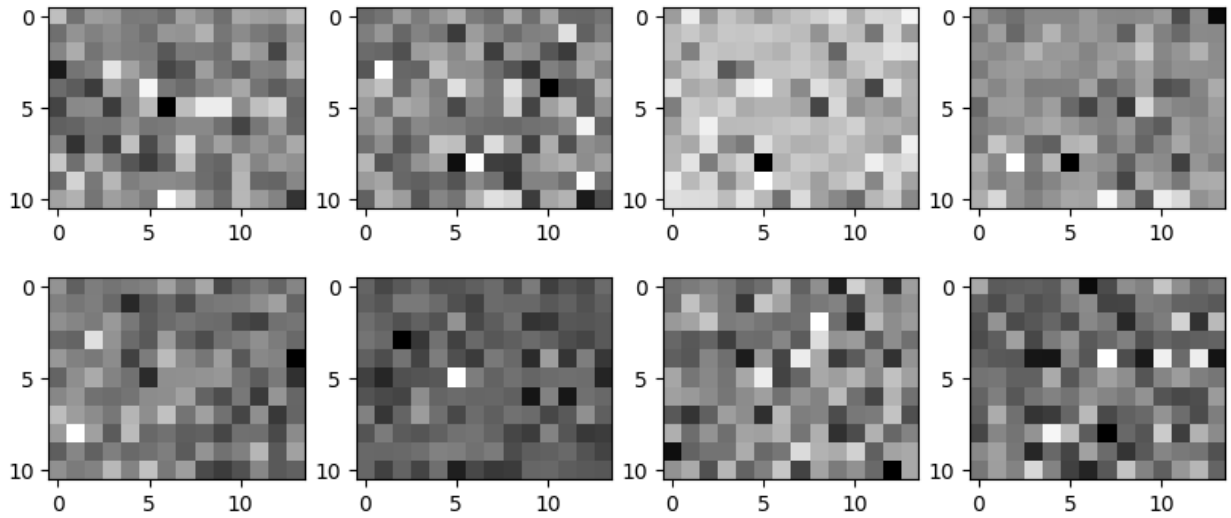
۰.۰۶۰۱۳۵۰۳	۰.۱۲۶۰۷۰۰۹	۰.۱۷۱۳۱۵۷۲	۰.۱۹۰۷۱۲۴۲

و ۴ مقدار ویژه آخر را می‌توان در ادامه دید:

$۲.۵۱۱۷۰۱۱۷ * ۱۰^{-۴}$	$۲.۴۶۹۴۲۴۰۸ * ۱۰^{-۴}$	$۲.۰۴۵۰۳۲۶۹ * ۱۰^{-۴}$	$۱.۷۲۸۹۴۷۳۷ * ۱۰^{-۳۰}$

همانطور که مقادیر ویژه فوق که معادل با واریانس تصاویر داده‌ها بر بردار ویژه متناظر آن‌هاست نشان می‌دهند، ۴ مقدار ویژه اول واریانس بسیار قابل توجهی دارند و به نظر می‌رسد که در صورت استفاده از این ابعاد مدل قدرت تفکیک بسیار زیادی بین دسته‌های مختلف داده‌ها داشته باشد. از طرفی دیگر، ۴ مقدار ویژه آخر دارای واریانس بسیار کمی هستند که نه تنها استفاده از آن‌ها برای تفکیک تصاویر مفید نیست، بلکه در صورت استفاده از این ابعاد در کنار ابعاد دیگر، قدرت تفکیکی به مدل نمی‌افزایند و تنها بر افزایش پیچیدگی مدل و در نتیجه هزینه زمانی و منابع تاثیر می‌گذارند.

همچنین تصاویر زیر نیز با استفاده از این مقادیر ویژه به دست آمده‌اند که به صورت خاصی قابل تفسیر نیستند. ردیف اول مربوط به ۴ مقدار ویژه اول و ردیف دوم مربوط به ۴ مقدار ویژه دوم هستند.



بخش دوم

در این سوال به پیاده‌سازی الگوریتم LDA به کمک کتابخانه sklearn می‌پردازیم.

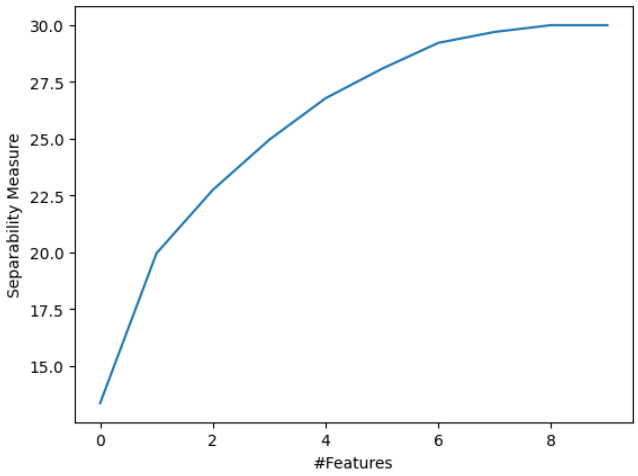
الف

پس از پیاده‌سازی LDA و fit کردن بر روی داده‌های موجود(Fashion-MNIST)، با توجه به ۱۰ کلاسه بودن مجموعه داده مورد نظر، تعداد ابعاد ایجاد شده توسط LDA برابر با ۹ است و مقادیر ویژه این ۹ بعد به ترتیب نزولی از چپ به راست به شرح زیر است:

۰.۴۴۵۶۴۹	۰.۲۱۹۷۹۰	۰.۰۹۳۰۴۴	۰.۰۷۳۴۱۵	۰.۰۶۰۹۴	۰.۰۴۳۲۲۹	۰.۰۳۷۹۸۸	۰.۰۱۶۰۲۱	۰.۰۰۹۹۱۱

ب

تابعی برای محاسبه مقدار Separability Measure از صفر پیاده‌سازی شده است و برای تعداد ویژگی‌های ۱ تا ۹ روی ویژگی‌های جدید دادگان موجود اعمال شده است. نتیجه این تابع به شکل زیر است:



با توجه به اینکه هر چه این معیار دارای مقدار بیشتری باشد، فاصله بین کلاس‌ها نیز در فضای ویژگی بیشتر است، در طول فرآیند مشاهده می‌کنیم که تعداد ویژگی بیشتر به صورت مداوم باعث افزایش این فاصله می‌شود اما شیب این نمودار از تعداد مشخصی از ویژگی‌ها به بعد کم می‌شود و با توجه به هزینه اضافی تحمیل شده در ازای افزایش ناچیز فاصله بین دادگان کلاس‌ها، به نظر می‌رسد در تعداد ویژگی ۶ که نمودار شکست پیدا کرده است و شیب آن بسیار کم شده است، شاهد بهترین توازن بین هزینه و فاصله بین دادگان کلاس‌ها خواهیم بود. بنابراین تعداد ویژگی پیشنهادی برابر با ۶ است که شامل ۶ ویژگی با بیشترین مقدار ویژه می‌شود.

سوال ۶

در این سوال ابتدا به بارگزاری مجموعه داده مربوطه می‌پردازیم و ویژگی‌هایی از تصاویر استخراج می‌کنیم. دسته اول ویژگی‌ها شامل میانگین دو بعد قرمز و آبی هر تصویر هستند و دسته دوم ویژگی‌ها شامل میانگین و مد بعد قرمز تصاویر است و نتایج روی هر دسته از ویژگی‌ها در ادامه قابل مشاهده است.سپس الگوریتم GMM را به کمک کتابخانه sklearn پیاده‌سازی می‌کنیم.

الف

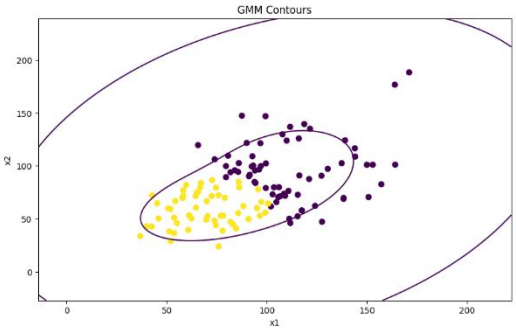
با استفاده از کد زیر یک GMM بر روی دادگان آموزش داده شده‌اند:

```
gm = GaussianMixture(n_components=2, random_state=0).fit(X)
```

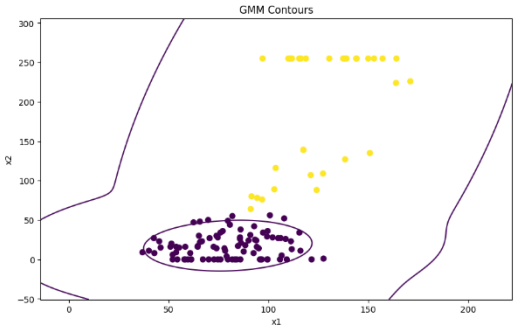
پس از آموزش، مقدار پارامترها و مقدار ضریب سیلوئت مدل‌های آموزش دیده به شرح زیر است:

	Mean	Weights	Score
دسته اول ویژگی‌ها	[[106.81314379, 91.7794563], [73.22719697, 62.12078031]]	[0.55833772, 0.44166228]	0.371
دسته دوم ویژگی‌ها	[[79.38927623, 17.16193388], [125.31515048, 184.48694753]]	[0.72585777, 0.27414223]	0.696

و نمودار داده‌ها به همراه کانتورهای مدل‌های آموزش دیده به شکل زیر است:



دسته اول ویژگی‌ها

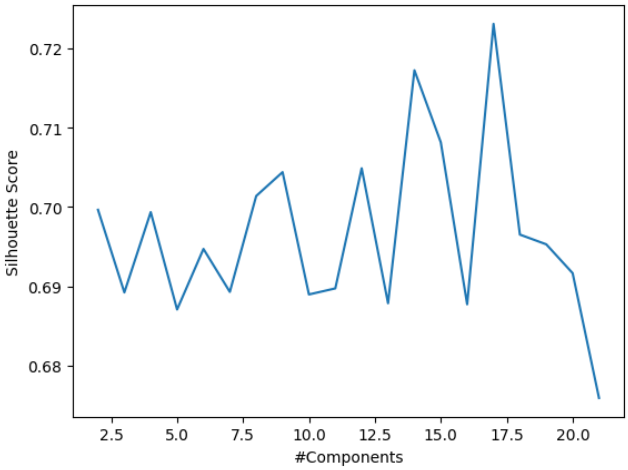


دسته دوم ویژگی‌ها

همانطور که مشخص است، مدل‌هایی که با استفاده از دسته دوم ویژگی‌ها(میانگین و مد بعد قرمز) آموزش دیده‌اند، عملکرد بهتری در ضریب سیلوئت و کانتورهای رسم شده از خود نشان داده‌اند.

ب

تعداد کامپوننت‌های ۲ تا ۲۲ به روش Cross_validation و با معیار ارزیابی ضریب سیلوئت ارزیابی شدند و امتیازات هر کدام به در نمودار زیر قابل مشاهده است:



همانگونه که مشخص است، بهترین مقدار ضریب سیلوئت در نقطه‌ای به نمایش درآمده است تعداد کامپوننت‌ها برابر با ۱۷ در نظر گرفته شده است اما این اختلاف آنقدر کم است که با توجه به تصادفی بودن ذاتی الگوریتم GMM، ممکن است در اجرایی دیگر این نتایج حاصل نشود و در نقطه‌ای دیگر شاهد بهترین عملکرد باشیم. با توجه به نتایج ظاهر شده به نظر می‌رسد که ویژگی‌های استخراج شده از تصاویر توان تفکیک پذیری زیادی را به مدل GMM نمی‌دهند و بنابراین تفاوت چندانی در تعداد مختلف کامپوننت‌ها شاهد نیستیم.