درس : مباحث ویژه در داده‌کاوی
دانشجو : امیرمحمد خرازی
شماره دانشجویی : ۴۰۱۵۲۵۲۱۰۰۲
استاد درس : دکتر منصور رزقی آهق

تمرین اول — دانشکده علوم ریاضی ، گروه علوم کامپیوتر، گرایش داده‌کاوی

گیت‌هاب درس (لینک) — گیت‌هاب این تمرین (لینک)

پرسش ۱ :

نشان دهید k-means یک روند کاهشی است.

پاسخ :
الگوریتم k-means بطور خلاصه به شکل زیر عمل می‌کند : ابتدا ما تعداد k را برای این الگوریتم مشخص می‌کنیم (روش‌هایی نیز وجود دارند که خودشان این k را پیدا می‌کنند). الگوریتم k-means مورد نظر ما، همان الگوریتمی است که در کتاب [۱] صفحه ۳۷۲ معرفی شده است. بعد از مشخص کردن این k، الگوریتم k مرکز تصادفی تولید می‌کند ($t = 0$) . در حال حاضر تعداد k تا خوشه داریم که خالی هستند (مراکز خوشه ها می‌توانند نقاطی فرضی باشند که در نمونه‌های واقعی ما وجود ندارند). سپس در هر گام دو کار انجام می‌دهیم :

۱. هر داده را به یک خوشه ارتباط می‌دهیم :

برای اینکار ابتدا **فاصله** هر داده تا مرکز خوشه مرحله قبل را محاسبه می‌کنیم. سپس داده را به خوشه‌ای متعلق می‌کنیم که با مرکز آن در گام قبلی، **کمترین** فاصله را داشته باشد. یعنی بطور مثال اگر $k = 3$ باشد، برای هر داده در هر مرحله از الگوریتم، فاصله داده تا مرکز خوشه (در مرحله قبل) را محاسبه می‌کنیم ، یعنی در این مثال ۳ فاصله محاسبه می‌شود، سپس داده را به خوشه ای که نماینده آن کمترین فاصله را با داده دارد، مرتبط می‌کنیم.

۲. مراکز جدید را برآورد می‌کنیم :

بعد از مشخص شدن تکلیف هر داده در این مرحله (در بخش قبل)، خوشه‌ها و اعضای آن‌ها مشخص هستند. کافی است نماینده خوشه ، که در اینجا ما آن‌را میانگین اعضای خوشه می‌گیریم، مشخص شود. برای اینکار کافی است از داده‌های عضو هر خوشه میانگین بگیریم و مرکز جدید خوشه را بدست آوریم.

مراحل بالا را تا زمانی که همرایی رخ دهد ادامه می‌دهیم. یعنی می‌توانیم یک شرط روی آن داشته باشیم که اگر مراکز خوشه‌ها تفاوت چندانی نکردند (در گام‌های متوالی)، آنگاه الگوریتم به بهینه خود رسیده است. این الگوریتم یک روش تکراری است که با مراکز خوشه تصادفی شروع به کار میکند و هر بار اعضای خوشه‌ها را بدست آورده و مراکز جدید را تشکیل می‌دهد تا به بهین برسد (یعنی مراکز تغییر زیادی نداشته باشند). اگر اشتباه نکنم، بهینه k-means خیلی خوب نیست و انتخاب نقاط تصادفی اولیه برای مراکز روی آن تاثیر دارد ، لذا چندین بار این الگوریتم را اجرا میکنند و بهترین را به عنوان جواب در نظر می‌گیرند.
دو نکته در بالا حائز اهمیت است : فاصله و کمترین . همانطور که از تعریف و رنود الگوریتم k-means مشخص است، از آنجایی که ما فاصله را به عنوان معیاری برای شباهت و عدم شباهت داده‌ها مشخص کردیم، هر چه فاصله یک داده از هم دیگر

دور باشد، کمتر شبیه هستند و برعکس، هر چه فاصله کم باشد، بیشتر شبیه هستند. هدف ما این است که داده‌هایی که شبیه هم هستند در یک خوشه قرار بگیرند و داده‌هایی که شبیه نیستند از هم دور باشند. برای اینکار، مجبوریم حداقل فاصله را در نظر بگیریم. لذا این الگوریتم، یک روند کاهشی دارد بدین صورت که در هر مرحله تلاش می‌کند فاصله داده تا مرکز خوشه مورد نظرش را کمینه کند. لذا هدف بهینه سازی الگوریتم k-means کمینه کردن مجموع فاصله‌ها است یعنی :

$$C^{\star} = \min SSE(C)$$

که در این تابع هدف، SSE برابر Sum of Squared Error است و بصورت زیر محاسبه می‌شود:

$$SSE(C) = \sum_{i=1}^{k} \sum_{x_j \in C_i} distance(x_j, \mu_i) \qquad \mu_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$$

که این فاصله را می‌توانیم نرم ۱ ، نرم ۲ یا غیره در نظر بگیریم که در پرسش دوم این مسئله بررسی می‌شود.

پس بطور خلاصه، در الگوریتم k-means هدف پیدا کردن مجموعه خوشه است که عناصر شبیه به هم در یک خوشه باشند. شباهت هر عنصر با خوشه یعنی شباهت عنصر با نماینده خوشه که در k-means این نماینده، میانگین عناصر موجود در خوشه است. طبیعتا عناصری که به نماینده خوشه خود شبیه هستند، به یکدیگر نیز شبیه‌اند.

گفتیم که در هر مرحله از الگوریتم فاصله داده تا مرکز خوشه‌ها محاسبه می‌شود و داده را به خوشه‌ای می‌بریم که تا مرکز آن کمترین فاصله را داشته باشد. با این تفسیر در هر مرحله مرکز خوشه به داده‌های آن خوشه نزدیک‌تر می‌شود تا جایی که دیگر تغییری نمی‌کند. نزدیکتر شدن یعنی فاصله آن کاهش می‌یابد (ممکن است فاصله مرکز خوشه با یک داده در مرحله‌ای نسبت به مرحله قبلش بیشتر شود ولی مجموع این فاصله‌ها، یعنی فاصله هر داده تا مرکز خوشه‌اش، در هر مرحله نسبت به مرحله قبل کاهش می‌یابد).

لذا با توجه به برداشت من از سوال، روند کاهشی در الگوریتم k-means را می‌توان بصورت بالا توضیح داد.

پرسش ۲ :

روش k-means را با نرم یک و نرم l_p به ازای $p = 4$ بنویسید.

پاسخ :

با توجه به الگوریتم k-means که در کتاب [۱] صفحه ۳۷۳ آورده شده است، و همچنین توضیحاتی که در پرسش قبل آورده شده است، کافی است در هر مرحله، در هنگام محاسبه فاصله از نرم‌های گفته شده استفاده کنیم. یعنی در هر گام، در مرحله عضویت بخشیدن هر داده به یک خوشه، بجای محاسبه فاصله به روز $L2$ از روش‌های $L1$ یا L_p برای $p = 4$ استفاده کنیم. فرض کنید x و y هر دو متعلق به فضای D بعدی باشند ($x, y \in \mathbb{R}^D$) . آنگاه $distance(x, y)$ را با توجه به نرم‌های زیر،

بیان می‌کنیم :

$$L2-norm \Longrightarrow distance(x,y) = \sqrt[2]{\sum_{i=1}^{D}(x_i-y_i)^2}$$
$$L1-norm \Longrightarrow distance(x,y) = \sum_{i=1}^{D}|x_i-y_i|$$
$$L_p-norm(p=4) \Longrightarrow distance(x,y) = \sqrt[p]{\sum_{i=1}^{D}(x_i-y_i)^4}$$

البته در $L2$ و L_p به جای (x_i-y_i) باید قدر مطلق آن یعنی $|x_i-y_i|$ نوشته شود که چون توان ۲ یا توان ۴ باعث می‌شود جواب مثبت باشد (زیرا در حال محاسبه فاصله هستیم که فاصله همیشه مثبت است، یعنی نرم همیشه مثبت است) دیگر از قدر مطلق استفاده نکردیم.

در k-means معمولی که در فضای اقلیدسی از نرم ۲ برای محاسبه فاصله استفاده می‌کند بدین صورت عمل می‌کردیم:

۱. در مرحله $(t=0)$ قرار داریم. ابتدا k مرکز تصادفی تولید/انتخاب می‌کنیم.

۲. مرحله $t=t+1$ ، تعداد k مجموعه خوشه خالی تولید می‌کنیم، یعنی مجموعه $C_1,\ldots,C_k$ که همه آن‌ها خالی هستند.

۳. برای هر داده فاصله این داده تا مرکز هر خوشه را محاسبه کرده و خوشه‌ای که حداقل این فاصله را داشت، داده را دربرمی‌گیرد. به زبان الگوریتمی‌تر، اگر x_j داده مورد نظر ما باشد که می‌خواهیم تکلیفش را مشخص کنیم و i روی خوشه‌های ما گردش می‌کند، آنگاه :

$$j = \min_i \left\{ distance(x_j, center_i^{(t-1)}) \right\}$$
$$C_j = C_j \cup \{x_j\}$$

۴. با توجه به هر خوشه، مراکز جدید را می‌سازیم. پس خواهیم داشت : $center_i^{(t)} = update(center_i^{(t-1)})$

۵. مرحله ۲ تا ۴ را تکرار می‌کنیم تا به شرط همگرایی برسیم.

در الگوریتم بالا که الگوریتم k-means معمولیست، فاصله و نماینده خوشه‌ها می‌توانند نسبت به هم انتخاب شوند. در اینجا نماینده را مرکز خوشه در نظر گرفتیم . مرکز خوشه می‌تواند میانگین اعضای هر خوشه باشد اما لزومی نداریم که حتما از میانگین به عنوان نماینده خوشه استفاده کنیم؛ مثلا می‌توانیم از میانه استفاده کنیم.

در $L2$ انگار دنبال برآورد یک مدل گاوسی آمیخته هستیم . به نظرم با این تفسیر در $L1$ به دنبال برآورد یک مدل لاپلاس یا نمایی هستیم.

فرض کنیم داده‌های ما $x_1, x_2, \ldots, x_n$ از توزیع گاوسی آمده باشند. براساس رابطه Maximum Likelihood برای تخمین پارامترهای مدل داریم :

$$f(x_1,x_2,\ldots,x_n) = \frac{1}{\sqrt{2\pi\sigma^2}} \prod_{i=1}^{n} \exp\left(-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2\right)$$

که با log گرفتن آن را به جمع تبدیل کرده و ماکسیمم کردن آن با منفی کنار x_i به مینیمم سازی تبدیل می‌شود که در نهایت مسئله را به کمینه کردن مجموع $(x_i - \mu)^2$ تبدیل می‌کند که این خودش همان مفهوم $L2$ است. به زبان دیگر :

$$\max\{-(\dots)\} \longleftrightarrow \min(\dots) \longleftrightarrow \min \sum_{i=1}^{n} (x_i - \mu)^2$$

این n در حقیقت برای یک خوشه است. حالا اگر توزیع نمایی باشد بطور مشابه داریم :

$$f(x_1, x_2, \dots, x_n) = \frac{1}{\beta^n} \prod_{i=1}^{n} \exp\left(\frac{x_i - \mu}{\beta}\right)$$

که بطور مشابه خواهیم داشت :

$$\max\{-(\dots)\} \longleftrightarrow \min(\dots) \longleftrightarrow \min \sum_{i=1}^{n} |x_i - \mu|$$

در موارد بالا منظور از ... ها در مینیمم سازی یا ماکسیمم سازی عباراتی هستند که بعد از log گرفتن بدست آمده‌اند که تعداد زیادی از آن‌ها ثابت هستند و تاثیری در بهینه سازی ندارند لذا ما فقط جملاتی را در نظر می‌گیریم که بهینه‌سازی در آن‌ها موثر است.

با این تفاسیر می‌توان $L2$ و $L1$ را برای k-means تا حدی تصور کرد. در حالت L_4 یعنی زمانی که از نرم ۴ استفاده می‌کنیم مثل تابع گاوسی است که $((x_i - \mu)^2)^2$ شده است. یعنی اینگار آن را به توان ۲ رسانده ایم. این موارد شاید با تعابیر از آمار مثلا اگر متغیر تصادفی X دارای توزیع گاوسی با پارامترهای ۱ ، ۱ باشد، متغیر تصادفی X^2 دارای چه توزیعی است؛ قابل درک باشد. چون ممکن است حجم پاسخ به این سوال بیش از حد انتظار شود، بیشتر از این وارد مطلب نمی‌شوم.

لذا بطور خلاصه کافی است برای استفاده از نرم‌های دیگر، فاصله را با آن نرم‌ها در الگوریتم k-means حساب کنیم و چنانچه لازم دیدیم، از نماینده‌های بهتری برای خوشه‌ها استفاده کنیم. بصورت پیشفرض نماینده هر خوشه مرکز آن یا میانگین اعضای آن خوشه می‌باشد که بصورت $\mu_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$ محاسبه می‌شوند.

نمونه‌های جدیدی از k-means برای این منظور ابداع شده‌اند :

- برای $L1$ ، K-medians ابداع شده است که مراکز را بجای میانگین، با میانه حساب می‌کند و خطا را برای زمانی که از $L1$ استفاده می‌کنیم، حداقل می‌کند (نسبت به زمانی که با میانگین و $L1$ استفاده می‌کردیم) .
- برای مترهای دیگر (دلخواه و احتمالا L_4) ، k-medoids ابداع شده است، این روش به نسبت k-means معمولی که از فاصله اقلیدسی $L2$ استفاده می‌کرد، پرقدرت تر است و به نویز و داده‌های پرت کمتر حساس است. عملا این روش یک کلی‌تر از k-means است.

روش این الگوریتم‌ها نیز نسبتا ساده است :

- برای k-medians ، کافی است با $distance$ ای که قبل‌تر معرفی کردیم (برای $L1$) و میانه داده‌ها عمل کنیم. برای میانه داده‌ها را مرتب کرده و آن‌هایی که وسط داده‌ها قرار دارند را به عنوان میانه می‌گیریم. اگر داده‌ها وسط نداشت، میانگین دو داده‌ها که در وسط هستند را به عنوان میانه می‌گیریم.

- برای k-medoids : یک نقطه واسط یا medoids ،عضوی از یک خوشه است که میانگین عدم شباهت آن با بقیه اعضای خوشه، کمینه است (یعنی وسط‌ترین است).

۱. ابتدا k نمونه را به عنوان نقاط واسط در نظر می‌گیریم (در قبل به صورت تصادفی k میانگین را به عنوان مرکز تولید می‌کردیم). لذا در اینجا مراکز اعضای واقعی هستند، یعنی خوشان از نمونه‌ها هستند.

۲. هر داده را به نزدیک ترین نقطه واسط (medoid) مرتبط می‌سازیم.

۳. برای هر داده‌ای که medoid نیست، فاصله اش را با medoid خوشه‌اش حساب می‌کنیم (این مقدار را مثلا $cost(m_i, o_j)$ برای خوشه i ام و داده j ام در نظر بگیرید.) . مجموع این هزینه‌ها (فاصله‌ها) را $cost$ می‌نامیم.

۴. نقطه دیگری را به عنوان نمونه واسط انتخاب می‌کنیم و هزینه را دوباره برای آن محاسبه می‌کنیم. برای مطلوب است که هزینه در هر گام، کمتر از گام قبلی باشد. اگر کمتر شد، آن را به عنوان نقطه واسط در نظر می‌گیریم.

۵. گام‌های ۲ تا ۴ را ادامه می‌دهیم تا الگوریتم همگرا شود.

طبیعی است که در هر نوع الگوریتم، از هر نوع متری برای اندازه‌گیری فاصله (نزدیکی) استفاده کنیم، از همان متر برای بهینه‌سازی نهایی الگوریتم استفاده خواهیم شد.

پرسش ۳ :

روش EM را با توزیع لاپلاس (به جای توزیع گاوسی) بنویسید و الگوریتم آن را با جزئیات بنویسید.

پاسخ :
با توجه به کتاب [۱] صفحه ۳۸۱ ، معادله (13.6) با توجه به توزیع لاپلاس نوشته می‌شود.

$$f(x, \mu_i, b_i) = \frac{1}{2b} \exp\left(-\frac{|x - \mu_i|}{b}\right)$$

معادلات بعدی آن (13.7) ، (13.8) و (13.9) شبیه کتاب است. هر بار پارامترهای مدل برآورد می‌شود تا به بهین برسیم که همان MLE است. روند آن شبیه به همان گاوسی است ولی در جا هایی که مثلا در تخمین واریانس از $(x - \mu_i)^2$ استفاده می‌کردیم، در اینجا به $|x - \mu_i|$ تبدیل می‌شود.
متاسفانه برای حل این سوال، خیلی فرصت کافی نداشتم و لذا پاسخ کاملی برای این سوال ارائه نداده‌ام.

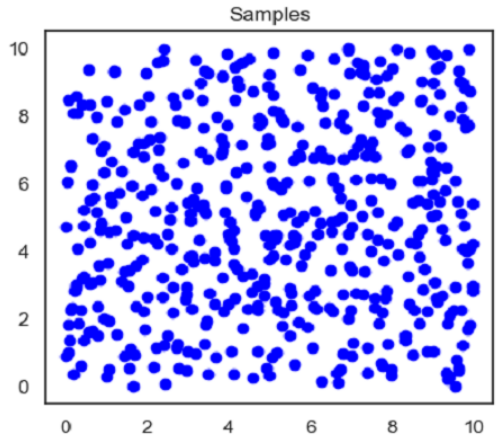
پرسش ۴ :

بین بازه‌ی $[0, 10] \times [0, 10]$ نزدیک ۵۰۰ تا داده رندوم به صورت یکنواخت تولید کنید. سپس این داده‌ها را توسط الگوریتم‌های زیر خوشه‌بندی کنید.

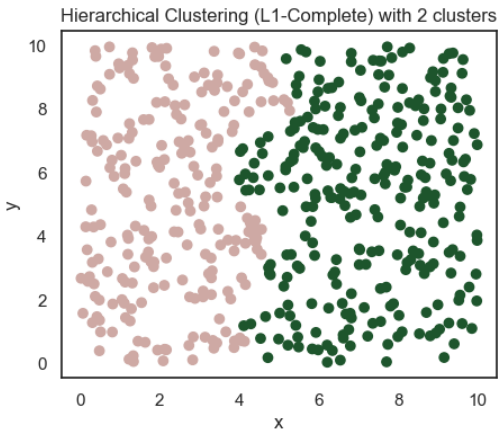
- روش سلسله مراتبی با نرم یک و Complete-link
- روش EM
- روش DBSCAN

نتایج این روش‌ها را تحلیل کنید.

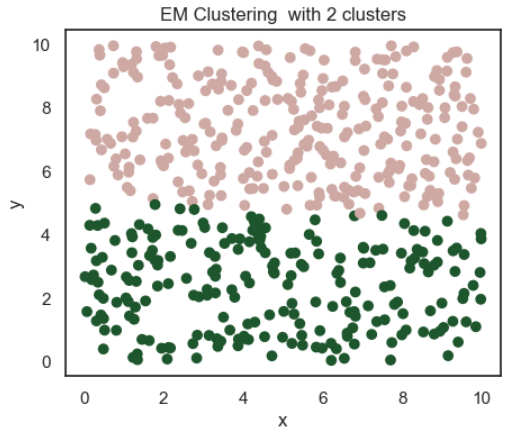
پاسخ :
جزئیات کامل کد نویسی این سوال، در کدهای ارائه شده قابل مشاهده و بررسی می‌باشد. کدهای این بخش را می‌توانید با عنوان `Uniform Random Numbers Clustering` در پوشه مربوطه بیابید. همچنین این کدها در لینک‌هایی که در ابتدای این گزارش ارائه شده است نیز موجود می‌باشد.
فرض کنید اگر تعداد خوشه‌ها برابر ۲ باشد، نتایج زیر بدست می‌آید:



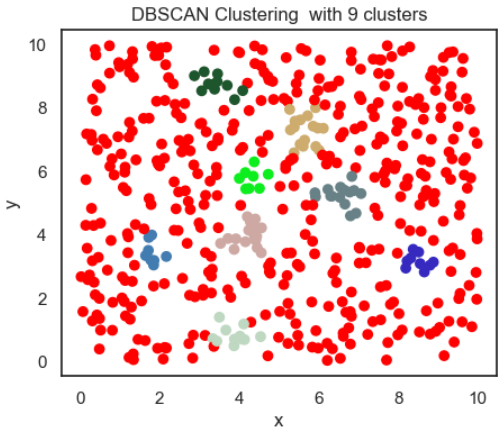
(آ) ۵۰۰ نمونه تصادفی



(ب) خوشه‌بندی با روش سلسه مراتبی نرم ۱ و complete-link



(ج) خوشه‌بندی با روش EM



(د) خوشه‌بندی با روش DBSCAN با مقدار $\epsilon = 0.5$ و $min_samples = 8$

پرسش ۵ :

داده‌های IRIS را دانلود کنید.

- این داده‌ها را با استفاده از الگوریتم‌های k-means ، Hierarchical ، EM و DBSCAN خوشه‌بندی کنید.
- این خوشه‌ها را توسط روش‌های مطرح شده (در کتاب قسمت supervised/External) ارزیابی کنید.

پاسخ :

جزئیات کامل کد نویسی این سوال، در کدهای ارائه شده قابل مشاهده و بررسی می‌باشد. کدهای این بخش را می‌توانید با عنوان `IRIS Clustering` در پوشه مربوطه بیابید. همچنین این کدها در لینک‌هایی که در ابتدای این گزارش ارائه شده است نیز موجود می‌باشد.

نتایج ارزیابی :

Results :

• Hierarchical Clustering

rand score : 0.8797315436241611

Normalized Mutual Information score : 0.7906785790830966

Fowlkes-Mallows score : 0.8237641241035158

• EM Clustering

rand score : 0.9574944071588367

Normalized Mutual Information score : 0.8996935451597475

Fowlkes-Mallows score : 0.9355985958131776

• DBSCAN Clustering

rand score : 0.7762863534675615

Normalized Mutual Information score : 0.5842137354876208

Fowlkes-Mallows score : 0.6887754949218047

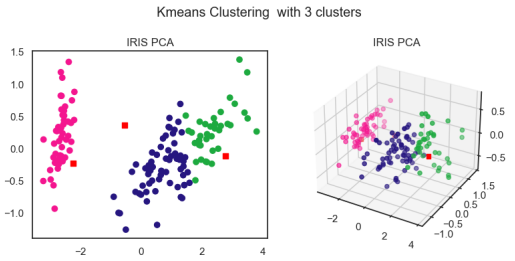
• Kmeans Clustering

rand score : 0.8797315436241611 Normalized Mutual Information score : 0.7581756800057784

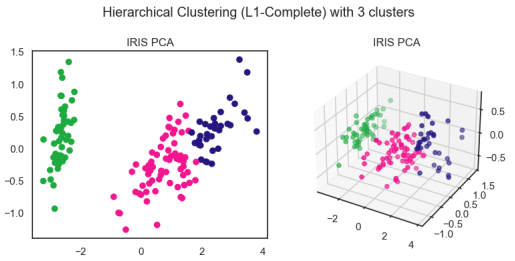
Fowlkes-Mallows score : 0.8208080729114153

مشاهده می‌شود که در اینجا الگوریتم EM روی داده‌های آموزش (داده‌های مشاهده شده) از همه بهتر جواب داده است.

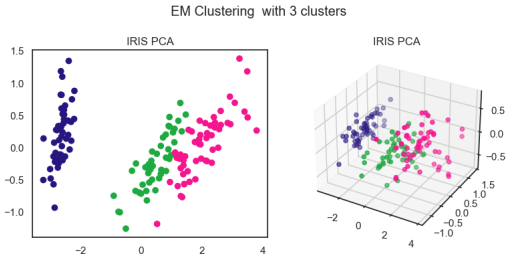
نتایج خوشه‌بندی بصورت زیر است :



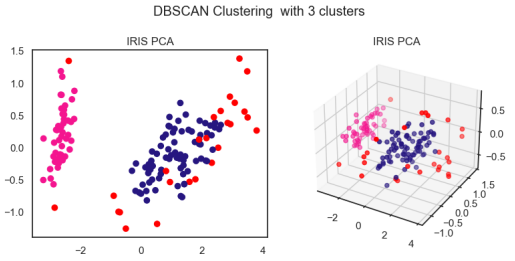
(آ) خوشه‌بندی با روش Kmeans با مقدار $\epsilon = 0.5$ و $min_samples = 8$



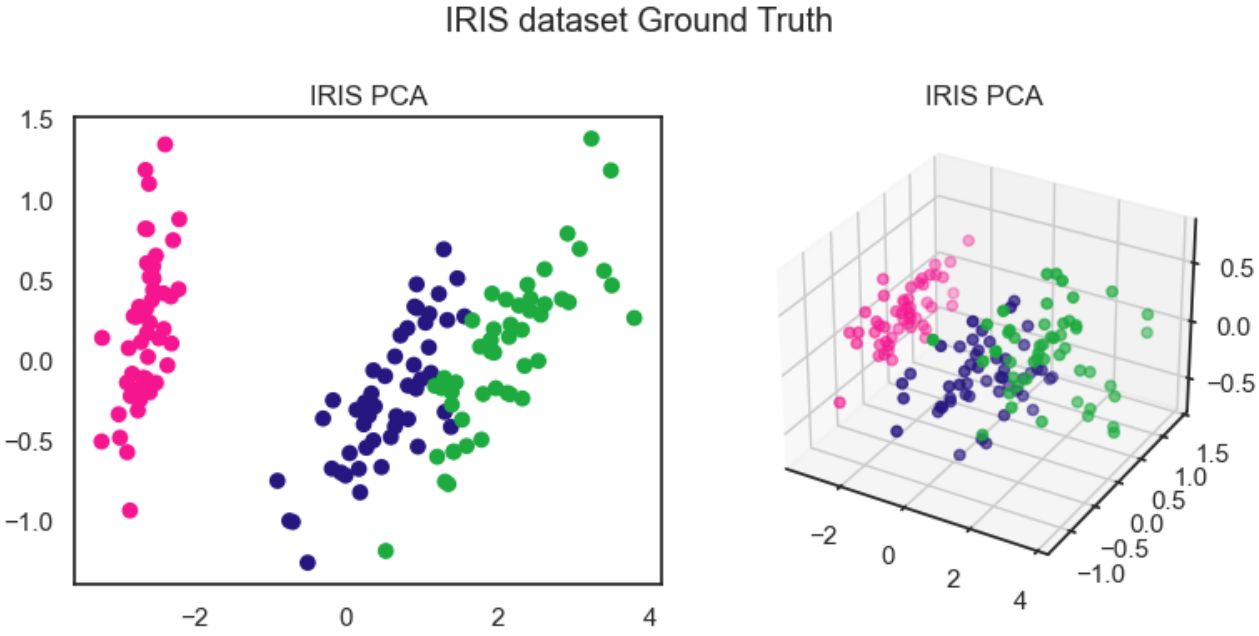
(ب) خوشه‌بندی با روش سلسه مراتبی نرم ۱ و complete-link



(ج) خوشه‌بندی با روش EM



(د) خوشه‌بندی با روش DBSCAN با مقدار $\epsilon = 0.5$ و $min_samples = 8$



(ه) Ground Truth داده‌های IRIS

مراجع

[1] Mohammed J Zaki, Wagner Meira Jr, and Wagner Meira. *Data mining and analysis: fundamental concepts and algorithms*. Cambridge University Press, 2014.