حل تمرین درس یادگیری ماشین

سری ۵

نام و نام خانوادگی دانشجو:

همایون حیدرزاده (۹۵۱۳۱۰۷۰)

نام استاد: دکتر ناظرفرد

بهمن ۹۶

۱)

مزیت‌های موارد ذکر شده در زیر آورده شده است:

میانگین:

برای داده‌های عددی مناسب می‌باشد. مزیت این معیار این است که مجموعه مجذور فاصله نقاط تا خود را کمینه می‌کند. میانگین نقطه‌ای فرضی است و از داده‌های مسئله نیست. این معیار به داده پرت حساس است و به سمت آن متمایل می‌شود و همچنین برای داده‌های غیر عددی مناسب نیست.

میانه:

برای داده‌های غیرعددی مناسب است. از میان داده‌های خوشه انتخاب می‌شود، بنابرین در مقابل داده پرت مقاوم‌تر است. این معیار نقطه‌ای با کمترین مجموعه فاصله از سایر نقاط را به عنوان مرکز برمی‌گزینند به صورت زیر محاسبه می‌شود:

$$x_{\text{medoid}} = \text{argmin}_{y \in \{x_1, x_2, \cdots, x_n\}} \sum_{i=1}^{n} d(y, x_i).$$

مد:

برای داده‌های غیرعددی مناسب است. در این روش داده با بیشترین بسامد به عنوان مرکز انتخاب می‌شود. این معیار برای داده‌های غیرعددی با عناصر تکراری مناسب است.

---

۲)

اگر ویژگی‌های مسئله واریانس متفاوت داشته باشند، مانند قیمت آپارتمان و تعداد اتاق‌ها. آن‌گاه روش‌های خوشه‌بندی معمول در جهت واریانس کمتر داده‌ها تمایل کمتری برای جدا شدن دارند و در جهت واریانس بالاتر داده‌ها بیشتر جداسازی خواهند شد. بنابرین مانند این است که برای ویژگی با واریانس کمتر ارزش بیشتری قائل شده ایم یا به عبارتی دیگر زمانی که معیار فاصله را حساب می‌کنیم ویژگی با تغییرات بیشتر تاثیر ویژگی با تغییرات کمتر را کاهش داده و روی نتایج خوشه‌بندی تاثیر بیشتری می‌گذارد.

بنابرین بهتر است قبل از استفاده از روش‌های خوشه‌بندی معمول همواره داده‌ها را نرمال کنیم.

---

۳)

در خوشه‌بندی سلسله مراتبی، برای محاسبه فاصله بین دو خوشه، از معیارهای زیر استفاده می‌شود:

link-Complete:

در این معیار برای محاسبه شباهت دو خوشه قطر دو خوشه پس از پیوند را به عنوان معیار در نظر می‌گیرند. قطر برابر بیشترین فاصله بین نقاط دو خوشه است. پیاده‌سازی شود دارای پیچیدگی زمانی O(n^2.Lg n) است. این الگوریتم به Outlier خیلی حساسیت بالایی نشان می‌دهد، زیرا وجود داده پرت می‌تواند قطر کلاستر نهایی را افزایش داده و این معیار را غیرمقاوم کند.

link-Single:

در این معیار نزدیک‌ترین فاصله بین دو خوشه را به عنوان معیار پیوند در نظر می‌گیرند. این فاصله برابر کمترین فاصله بین نقاط دو خوشه است. اگر این الگوریتم با استفاده از minimum spanning tree پیاده‌سازی شود دارای پیچیدگی زمانی O(n^2) است. این روش در بعضی مواقع نسبت به داده پرت مقاوم است ولی علاقه‌مند به تشکیل زنیجره‌های طولانی از خوشه‌ها دارد.

link-Average:

در این معیار میانگین فواصل تمامی زوج نقاط بین دو خوشه را به عنوان معیاری برای پیوند دو خوشه در نظر می‌گیرند. این معیار از linkage-Complete نسبت به داده پرت مقاوم‌تر و از linkage-Single علاقه‌کمتری به تشکیل زنجیره‌های طولانی از خوشه‌ها دارد. پیچیدگی زمانی این روش O(n^2.Lg n) است.

---

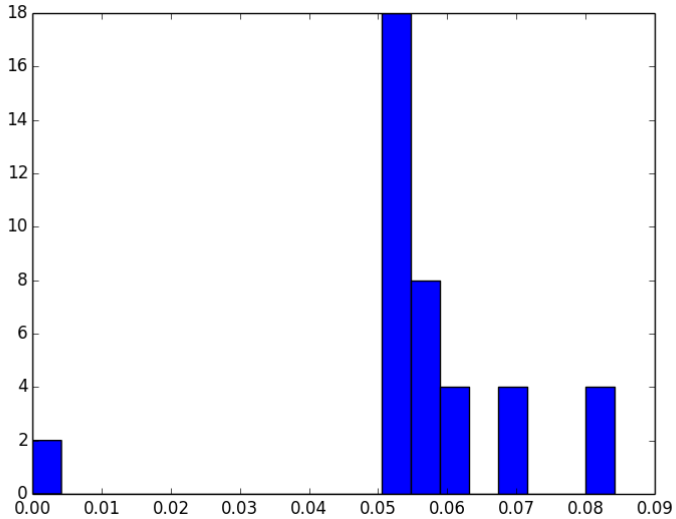
۴)

در اینجا روشی ارائه می‌شود که مبنای آن بر پایه کوچک کردن فضای جستجو و سپس جستجوی گرید در فضای به دست آمده است. می‌توان از سایر روش‌های جستجو مانند کلونی مورچگان و الگوریتم PSO برای جستجو سریع‌تر نیز استفاده کرد. مراحل کار با توجه به فراسنج‌ها به صورت زیر خواهد بود:

نکته: **این روش در سوال ۵ مورد استفاده قرار گرفته است و به جواب مناسب رسیده است.**

Epsilon:

هدف پیدا کردن نواحی مناسب برای جستجوی مقادیر epsion است. برای این کار ابتدا برای هر نقطه فاصله نزدیکترین همسایه‌اش را حساب نموده و ذخیره می‌کنیم. سپس میانگین فاصله‌ها را بر روی هیستوگرام برده و با این فرض که احتمالا ۱۰ تا ۲۰ درصد داده‌ها پرت باشند مقادیر epsilon را به صورت بازه‌ای حساب می‌کنیم. اگر تعداد داده‌ها خیلی زیاد باشد می‌توانیم از روش‌های نمونه برداری استفاده کنیم و یا فقط یک مقدار epsilon را برای مرحله بعد در نظر بگیریم. فرضا در نمودار هیستوگرام زیر ۰.۰۷ به عنوان یک مقدار کاندید برای epsilon انتخاب می‌شود زیرا حدود ۰.۹ داده فاصله کمتری از ۰.۷ دارند.

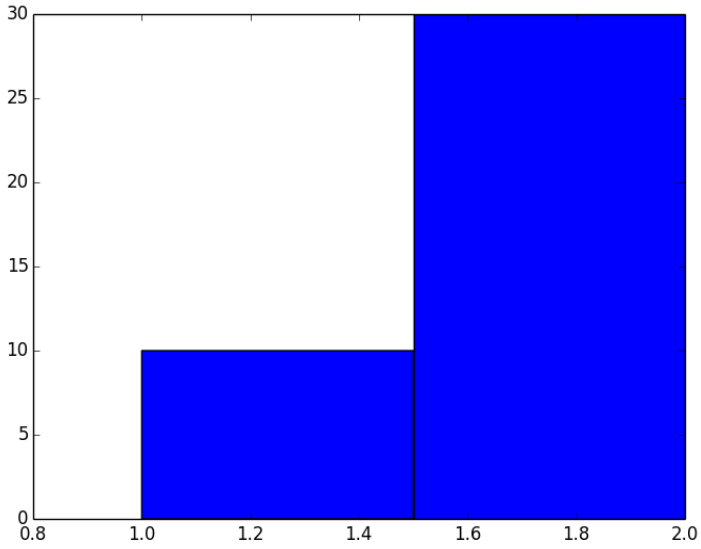


MinPoints:

در این مرحله برای epsilon به دست آمده در مرحله قبل به ازای تمامی داده‌ها مقادیر EpislonNeighbourhood را محاسبه کرده و هیستوگرام مجموعه داده حاصل را رسم می‌کنیم.

EpislonNeighbourhood برای یک نقطه برابر تعداد همسایه‌های آن نقطه در فاصله کمتر از epsilon آن نقطه است.

در هیستوگرام حاصل مشاهده خواهد شد که درصد کمی نقطه دارای همسایه‌های کم هستند. در الگوریتم dbscan پارامتر minPonits دارای حساسیت کمتری است بنابرین از یک جایی به بعد می‌توان تعداد همسایه را به عنوان minPoints مورد بررسی قرار داد و یا در نمودار هیستوگرام تعداد همسایه با فراوانی بالا را مورد بررسی قرار داد. فرضا در هیستوگرام زیر که مربوط به سوال ۵ است تعداد نقاط با همسایگی ۱.۵ تا ۲ بسیار



بیشتر هستند پس بهتر است الگوریتم dbscan را با minPointsهای ۱ و ۲ مورد بررسی قرار داد.

---

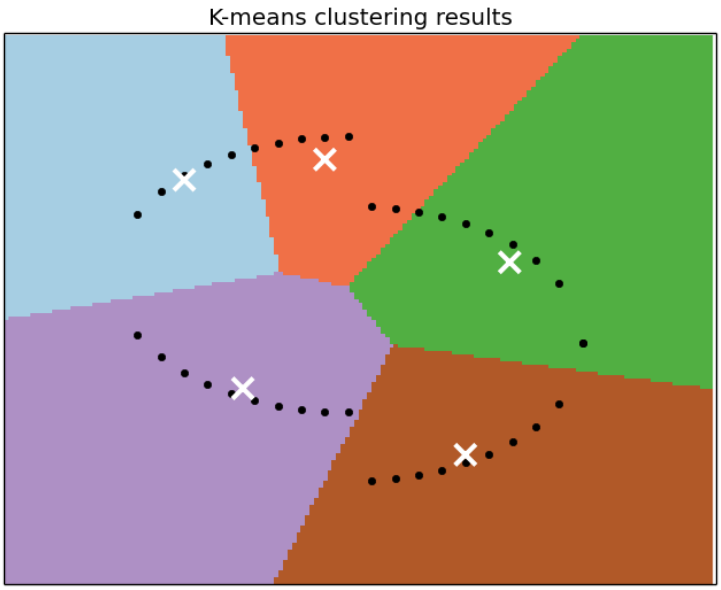
۵)

کد این قسمت در فایل main1.py و خروجی‌های کد در پوشه main1/outputs قرار گرفته است.

نکته: **داده‌ها قبل از استفاده در خوشه‌بندی نرمال شده‌اند.**

الف)

خروجی KMeans با استفاده از معیار DBI به صورت زیر به دست آمد:



همچنین بهترین k برای این مجموعه داده ۵ به دست آمد. مقادیر مختلف برای kهای مختلف در زیر آورده شده است:

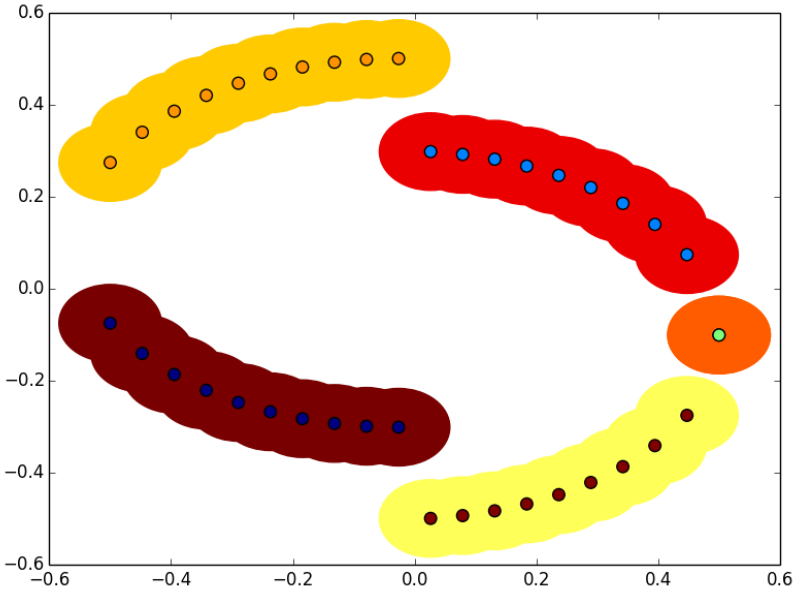
K	DBI
2	0.169496
3	0.109584
4	0.059035
5	0.051015
best(k)	5

ب)

کد این قسمت در فایل main1.py و خروجی‌های کد در پوشه main1/outputs قرار گرفته است.

نکته: **داده‌ها قبل از استفاده در خوشه‌بندی نرمال شده‌اند.**

با استفاده از روش پیشنهادی در سوال ۴ خوشه‌بندی به صورت زیر حاصل شد:



در نتایج به دست آمده در این خوشه بندی داده مقادیر فراسنج‌ها به صورت زیر به دست آمد:

Epsilon	Min points
0.084241	2

نکته حائز اهمیت این است که در اینجا به نظر می‌رسد داده تک سمت راست، داده‌ای پرت بوده است که الگوریتم آن را اشتباها یک خوشه در نظر گرفته است ولی در اصل این داده، دو داده است که روی هم افتاده است و الگوریتم چون شرط اپسیلون را همیشه ارضا می‌کند فقط شرط minPts می‌تواند بر روی آن تاثیر بگذارد.

ج)

خوشه‌های روش DBScan به صورت شهودی مناسب‌تر به نظر می‌رسند. بنابرین خوشه‌بندی قسمت ب بهتر عمل کرده است. به طور کلی خوشه‌بندی Kmeans برای داده‌هایی مناسب است که داده‌ها به شکل توزیعی شبیه توزیع نرمال باشند و خوشه‌ها به دست آمده همگی کانوکس هستند. بنابرین برای مجموعه داده غیر کانوکسی مانند داده‌های این مسئله، روشی مانند dbscan که می‌تواند هر نوع خوشه‌بندی را استخراج کند، مناسب‌تر است.

---

۶)

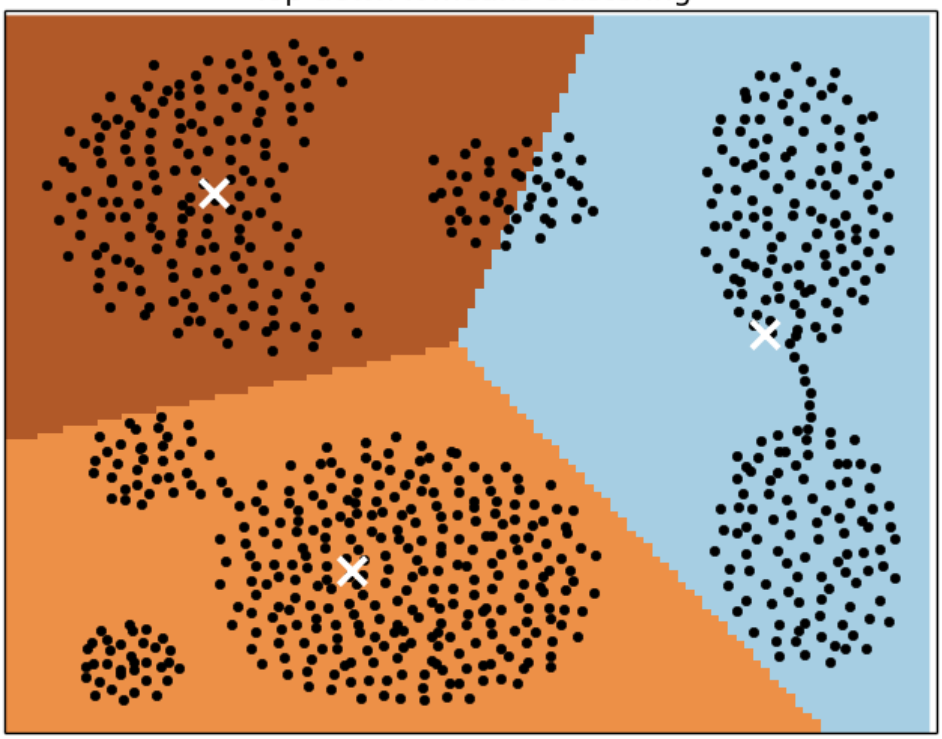
الف)

کد این قسمت در فایل py.main2 و خروجی‌های کد در پوشه main2/outputs قرار گرفته است.

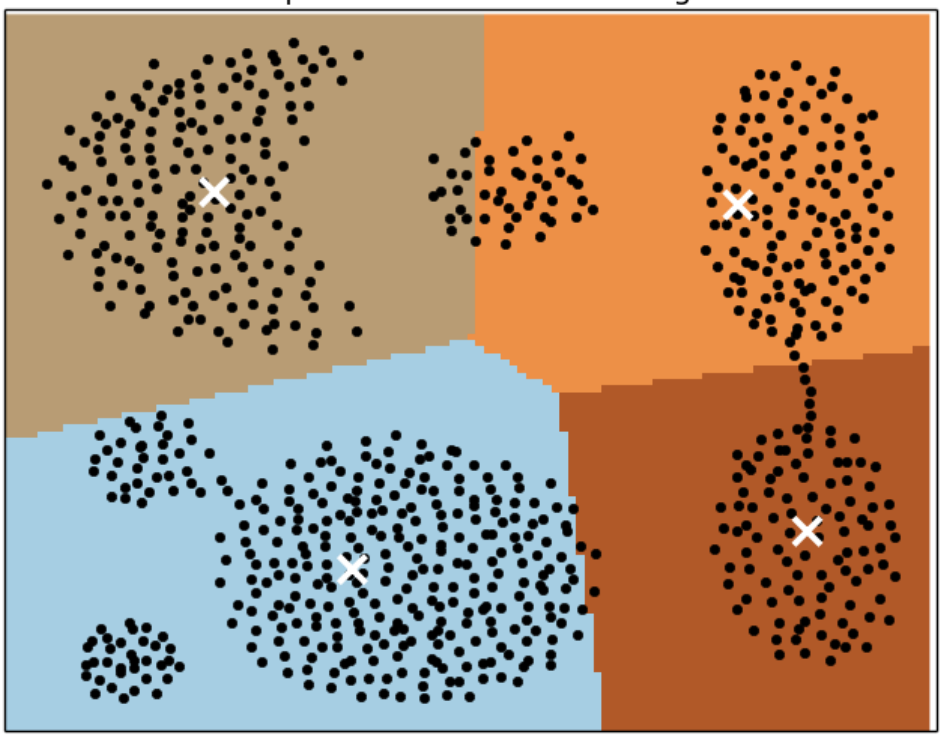
نکته: **داده‌ها قبل از استفاده در خوشه‌بندی نرمال شده‌اند.**

برای این قسمت الگوریتم Kmeans به صورت TopDown با $K=2$ پیاده‌سازی شد. مراحل انجام خوشه‌بندی از مراحل ۳ تا ۶ آورده است (سایر مراحل در پوشه خروجی متناظر موجود هستند)

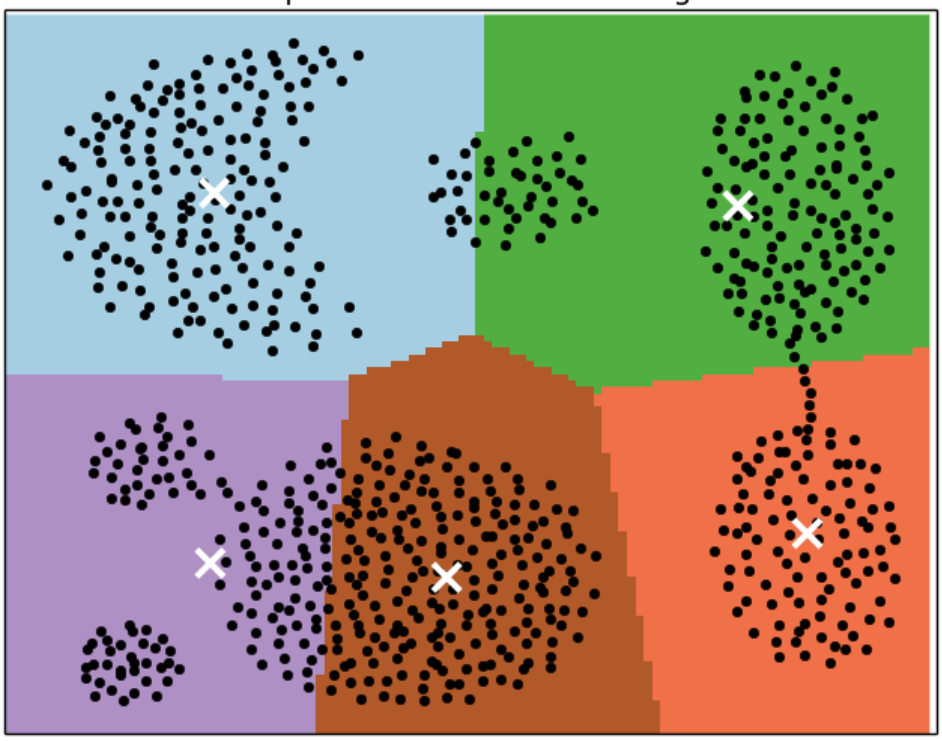
Top down K-means clustering



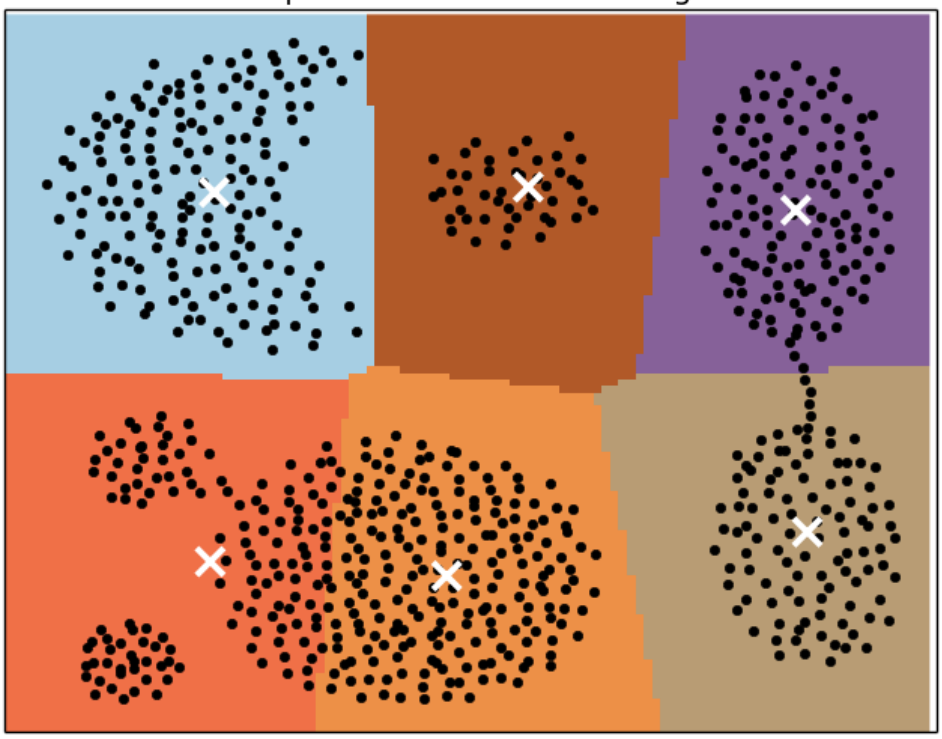
Top down K-means clustering



Top down K-means clustering



Top down K-means clustering

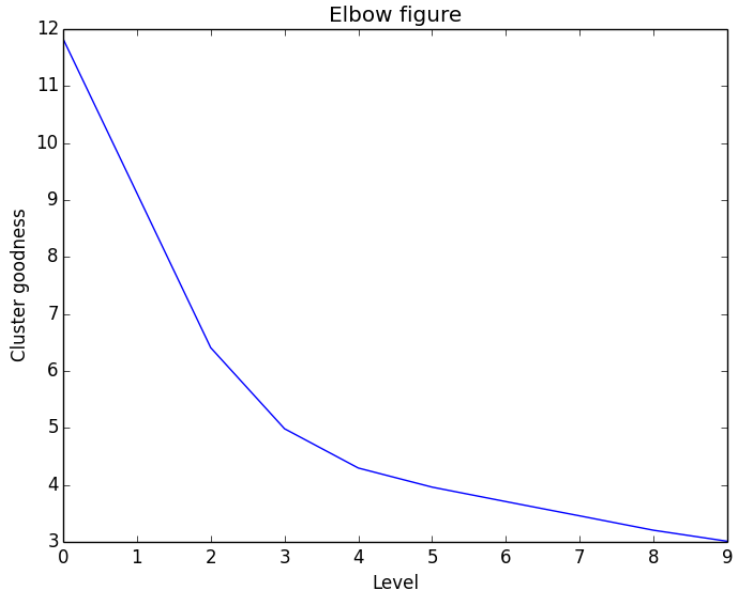


نتایج خوشه‌بندی در هر مرحله در جدول زیر آورده شده است:

Time	Loss(sse)	#Clusters
1	11.828174	1
2	9.120599	2
3	6.408474	3
4	4.986337	4
5	4.296626	5
6	3.961356	6
7	3.707511	7
8	3.457166	8
9	3.204756	9
10	3.011275	10

ب)

این قسمت با استفاده از الگوریتم elbow پیاده‌سازی شد. در این روش در هر مرحله درصد واریانس به صورتی تابعی از خوشه‌بندی نشان داده شده و طبق این قاعده زمانی که این تابع در یک نقطه دچار شکست شود(مانند آرنج)، آن نقطه به عنوان نقطه پایانی الگوریتم شناخته می‌شود. البته این روش می‌تواند به همراه شهود دیداری استفاده شود، در این سوال طبق این روش در مرحله سوم باید خوشه‌بندی متوقف شود:



اما به صورت شهودی مشخص است که مرحله ۶ نیز دارای خوشه‌بندی مناسب‌تری است.

ج)

کد این قسمت در فایل py.main3 و خروجی‌های کد در پوشه main3/outputs قرار گرفته است.

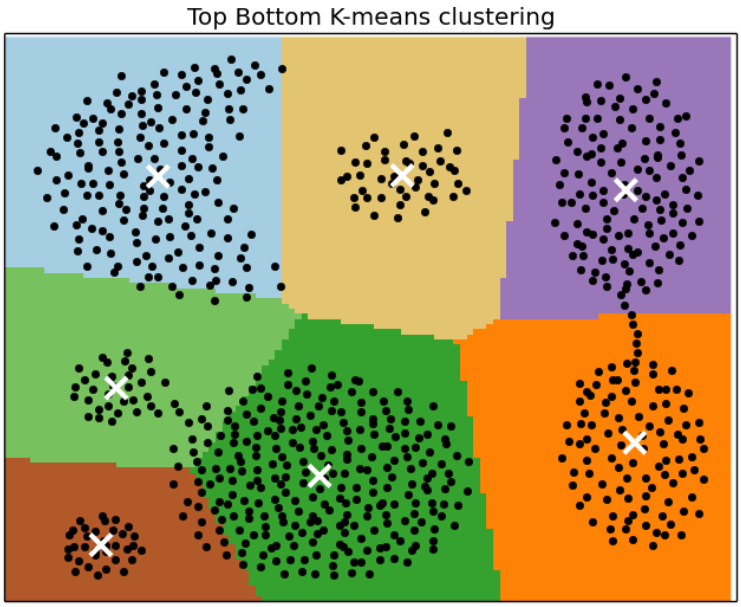
نکته:

☑ **داده‌ها قبل از استفاده در خوشه‌بندی نرمال شده‌اند.**

☑ **کدهای مربوط به این بخش با استفاده از روش برنامه‌نویسی پویا پیاده‌سازی شده‌اند.**

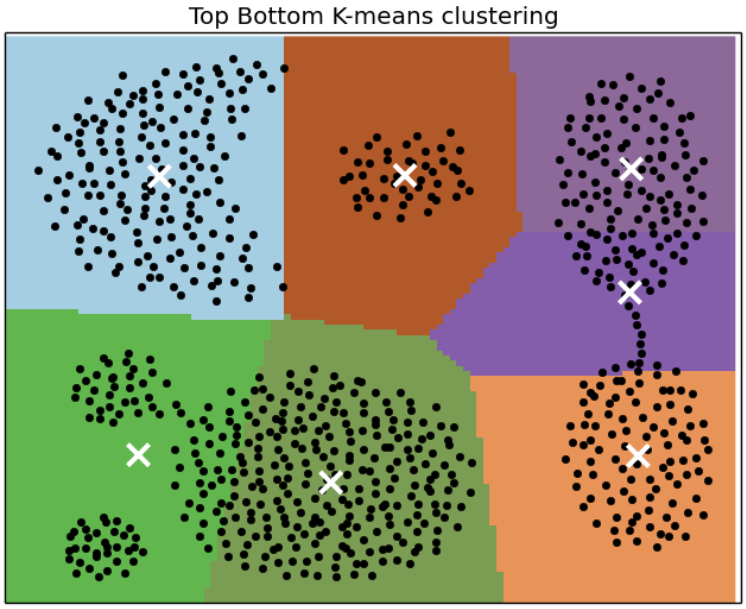
link-Average:

نتایج خوشه‌بندی در پوشه outputs/main3/avg موجود هستند و برای پرهیز از شلوغ شدن گزارش فقط نتیجه خوشی بندی در مرحله ۷ آورده شده است:



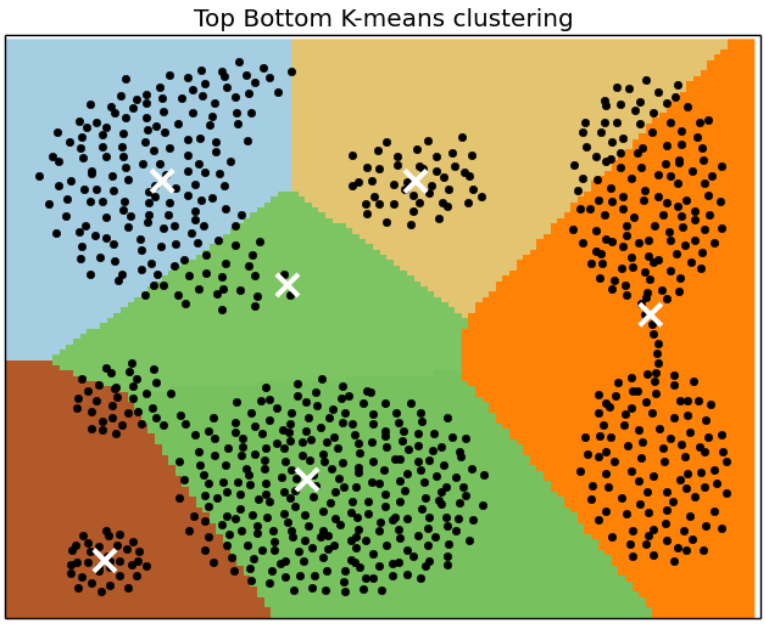
link-Complete:

نتایج خوشه‌بندی در پوشه outputs/main3/complete موجود هستند و برای پرهیز از شلوغ شدن گزارش فقط نتیجه خوشی بندی در مرحله ۷ آورده شده است.



link-Single:

نتایج خوشه‌بندی در پوشه outputs/main3/single موجود هستند و برای پرهیز از شلوغ شدن گزارش فقط نتیجه خوشی بندی در مرحله ۶ آورده شده است:



همان‌طور که انتظار می‌رفت link-Average بهترین نوع خوشه‌بندی را ارائه داده است. روش Complete به داده پرت حساس است و برای مثال در این روش خوشه بنفش (سمت راست وسط) به خاطر وجود داده پرت ایجاد شده است. همچنین در روش Single خوشه‌بندی بسیار نامنظم است زیرا در این روش پیوند خوشه‌ها معمولا به صورت زنجیروار ایجاد شده و بنابرین زمانی که داده‌های پرت وجود داشته باشند ممکن است خوشه‌هایی نا متعارف ایجاد شود (شکل بالا خوشه نارنجی).