به نام خدا

دانشگاه تهران

دانشکدگان فنی

دانشکده مهندسی برق و کامپیوتر

درس داده کاوی

تمرین عملی سوم

نام و نام خانوادگی : حسین سیفی

شماره دانشجویی : ۸۱۰۱۰۰۳۸۶

خرداد ۱۴۰۱

فهرست

تمرینات تشریحی

سوال ۱

در این سوال قصد داریم به ارزیابی خوشه بندی انجام شده در صورت سوال به کمک معیارهای ارزیابی متفاوتی از جمله Purity، entropy و Precision بپردازیم. داده‌های مربوط به این خوشه بندی در جدول زیر قابل مشاهده است:

	Entertainment	Financial	Foreign	Metro	National	Sport	Total
#1	1	1	0	11	4	676	693
#2	27	89	333	827	253	33	1562
#3	326	465	8	105	16	29	949
Total	354	555	341	943	273	738	3204

الف

در این بخش ابتدا مقدار Entropy را برای هر کدام از خوشه‌های خوشه‌بندی انجام شده به کمک فرمول زیر محاسبه می‌کنیم:

$$Entropy = -\sum_{i=1}^{r} P_{C_i} * \log_2 P_{C_i}$$

و برای این محاسبات از جدول زیر که شامل مقادیر Pc_i و $Pc_i * \log_2 Pc_i$ برای هر کلاس است استفاده شده است:

	#1		#2		#3	
	Pc_1	$Pc_1 * \log Pc_1$	Pc_2	$Pc_2 * \log Pc_2$	Pc_3	$Pc_3 * \log Pc_3$
Entertainment	0.001443	-0.01362	0.017286	-0.10119	0.343519	-0.52955
Financial	0.001443	-0.01362	0.056978	-0.23552	0.489989	-0.50429
Foreign	0	0	0.213188	-0.47537	0.00843	-0.05808
Metro	0.015873	-0.09488	0.529449	-0.48574	0.110643	-0.3514
National	0.005772	-0.04292	0.161972	-0.42537	0.01686	-0.09931
Sport	0.975469	-0.03495	0.021127	-0.11757	0.030558	-0.15378

$Entropy_1 = -(\text{-}0.01362\text{-}0.01362+0\text{-}0.09488\text{-}0.04292\text{-}0.03495) = 0.19999$

$Entropy_2 = -(\text{-}0.10119\ \text{-}0.23552\ \text{-}0.47537\ \text{-}0.48574\ \text{-}0.42537\ \text{-}0.11757) = 1.840748$

$Entropy_3 = -(\text{-}0.52955\ \text{-}0.50429\ \text{-}0.05808\ \text{-}0.3514\ \text{-}0.09931\ \text{-}0.15378) = 1.69641$

در نهایت برای محاسبه Entropy خوشه بندی می‌بایست میانگین وزن‌داری از Entropy خوشه‌ها را محاسبه کنیم.

$$Entropy_c = \left(\frac{693}{3204} * 0.19999\right) + \left(\frac{1562}{3204} * 1.840748\right) + \left(\frac{949}{3204} * 1.69641\right) = 1.443113$$

ب

در این بخش می‌توان ابتدا معیار Purity هر یک از خوشه‌های ایجاد شده را محاسبه کرد و سپس Purity کل خوشه‌بندی که میانگین وزن‌داری از Purity هر یک از خوشه‌ها هست محاسبه می‌شود اما روش ساده‌تر و کوتاه‌تر استفاده از فرمول زیر می‌باشد:

$$Purity = \frac{1}{n}\sum_{i=1}^{r} \max_{j=1\dots k}\{n_{ij}\}$$

و با توجه به اینکه :

$$\max_{j=1\dots 6}(n_{1j}) = 676$$

$$\max_{j=1\dots 6}(n_{2j}) = 827$$

$$\max_{j=1\dots 6}(n_{3j}) = 465$$

فرمول فوق به شکل زیر بر روی داده‌های جدول خوشه‌بندی اعمال می‌شود:

$$Purity_c = \frac{1}{3204}(676 + 827 + 465) = \frac{1959}{3204} = 0.614$$

ج

در این بخش به محاسبه معیارهای Precision، Recall و F-Measure می‌پردازیم. محاسبه مقدار معیار ارزیابی Precision برای هر کلاس به روش مشابهی با فرمول $p_i = \frac{1}{n_i}\max_{j=1\dots k}\{n_{ij}\}$ انجام می‌شود. همچنین معیار Recall به کمک فرمول $R_i = \frac{n_{ij_i}}{m_{j_i}}$ محاسبه می‌شود که با کمک این فرمول نشان می‌دهیم هر خوشه چند درصد از دسته داده‌های واقعی که نماینده آن شده است را پوشش می‌دهد. و در نهایت به محاسبه F_measure برای هر خوشه می‌پردازیم که یک میانگین هارمونیک از Precision و Recall است و به صورت $F = \frac{2}{\frac{1}{P}+\frac{1}{R}}$ محاسبه می‌شود.

$$P_1 = \frac{1}{693}\max\{1,1,0,11,4,676\} = \frac{676}{693} = 0.975$$

$$P_2 = \frac{1}{1562}\max\{27,89,333,827,253,33\} = \frac{827}{1562} = 0.529$$

$$P_3 = \frac{1}{949}\max\{326,465,8,105,16,29\} = \frac{465}{949} = 0.489$$

$$R_1 = \frac{676}{738} = 0.915$$

$$R_2 = \frac{827}{943} = 0.876$$

$$R_3 = \frac{465}{555} = 0.837$$

$$F_1 = \frac{2}{\frac{1}{P_1} + \frac{1}{R_1}} = \frac{2}{\frac{693}{676} + \frac{738}{676}} = \frac{1352}{1431} = 0.944$$

$$F_1 = \frac{2}{\frac{1}{P_1} + \frac{1}{R_1}} = \frac{2}{\frac{1562}{827} + \frac{943}{827}} = \frac{1654}{2505} = 0.660$$

$$F_1 = \frac{2}{\frac{1}{P_1} + \frac{1}{R_1}} = \frac{2}{\frac{949}{465} + \frac{555}{465}} = \frac{930}{1504} = 0.618$$

سوال ۲

در این سوال به خوشه‌بندی دانش‌آموزان بر اساس محل نشستن آن‌ها به کمک الگوریتم DBScan می‌پردازیم.

الف

دانش‌آموزان {C,E,F,H,I} از یک خوشه و دانش‌آموز {L} از خوشه‌ی دیگر جزو نقاط هسته‌ای هستند. هر کدام از این دانش‌آموزان ذکر شده حداقل ۴ همسایه با فاصله منهتن کمتر از ۲ واحد در اطراف خود دارند. در جدول زیر هر کدام از این دانش‌آموزان به همراه ۴ عدد از همسایگان متصل(Connected) و فاصله از آن‌ها ذکر شده‌اند.

Student	Neighbor 1	Neighbor 2	Neighbor 3	Neighbor 4
C	C:0	A:1	B:2	E:1
E	E:0	C:1	F:1	H:1
F	F:0	I:1	E:1	H:1
H	H:0	E:1	F:1	I:1
L	L:0	K:1	O:2	M:2
I	I:0	E:2	F:1	H:1

سایر دانش‌آموزان همسایه‌های کمتری در فاصله کمتر از ۲.۱ واحدی خود دارند که باعث می‌شود در بین نقاط هسته‌ای نباشند.

ب

دانش‌آموزان {A,B,G} از یک خوشه و دانش‌آموزان {K,M,O} از خوشه‌ی دیگر جزو نقاط مرزی هستند. این دانش‌آموزان به دانش‌آموزان هسته‌ای متصل هستند اما در فاصله ۲.۱ واحدی خود حداقل به تعداد minPts دانش‌آموزی را نمی‌بینند. در جدول زیر همسایگان هر یک از دانش‌آموزان مرزی به همراه فاصله آن‌ها از یکدیگر ذکر شده است و نقطه یا نقاط هسته‌ای که باعث پیوستن نقاط مرزی به خوشه‌ها شده است با رنگی متفاوت نمایش داده شده‌اند.

Student	Neighbor 1	Neighbor 2	Neighbor 3
A	C:1	E:2	-
B	C:2	-	-
G	F:2	-	-
K	L:1	J:2	-
M	L:2	N:1	-
O	L:2	-	-

سایر دانش‌آموزان {D,N,J} به هیچ یک از نقاط هسته‌ای متصل نیستند و در حالی که برخی از آنان با نقاط مرزی اتصال دارند اما باعث پیوستنشان به خوشه‌ها نمی‌شود و این سه دانش‌آموز جزو Outlierها محسوب می‌شوند.

ج

همانطور که در جداول فوق قابل مشاهده است، دانش‌آموزان H و F با فاصله ۱ و دانش‌آموز E با فاصله ۲ به طور مستقیم از دانش‌آموز I قابل دسترس هستند و فاصله‌ای کمتر از اپسیلون دارند.

د

همانطور که در جداول فوق قابل مشاهده است، دانش‌آموز L با فاصله ۲ و دانش‌آموز N با فاصله ۱ از دانش‌آموز M به طور مستقیم قابل دسترس هستند و فاصله‌ای کمتر از اپسیلون دارند. اگرچه دانش‌آموز N دسترسی مستقیم به دانش‌آموز M دارد اما با توجه به اینکه M یک نقطه مرزی است، نمی‌تواند نقطه N را به عضویت خوشه‌ی خود دربیاورد.

ه

جای مناسب برای دانش‌آموز P در حالتی که به هر دو خوشه متصل باشد اما باعث ادغام آن‌ها نشود در ستون ۳ و سطر ۵ می‌باشد. در این نقطه دانش‌آموز P به دو دانش‌آموز H و L متصل است(با فاصله منهتن ۲) اما به اندازه کافی همسایه متصل ندارد(۴ همسایه با فاصله کمتر از اپسیلون) تا P را تبدیل به یک نقطه هسته‌ای کند و باعث ادغام دو خوشه شود.

و

در صورتی که بخواهیم دانش‌آموز P باعث ادغام دو خوشه شود باید نقطه‌ای را انتخاب کنیم که حداقل ۳ همسایه با فاصله کمتر از ۲.۱ داشته باشد همچنین حداقل یک نقطه از هر خوشه با اضافه شدن دانش‌آموز P تبدیل به نقطه هسته‌ای شود. در جدول زیر نقاطی که شرایط فوق رو دارا هستند به همراه نقاط متصل ذکر شده‌اند و همسایگان هسته‌ای(پس از جای‌گیری P در نقطه ذکر شده) با رنگ پس زمینه متفاوت مشخص شده‌اند.(واضح است که همسایه چهارم خود دانش‌آموز P است و در جدول ذکر نشده است.)

Column	Row	Neighbor 1	Neighbor 2	Neighbor 3
4	6	L:2	H:2	M:2
5	6	M:1	N:2	I:2
5	5	H:2	I:1	M:2

با توجه به جدول فوق ۳ نقطه مناسب برای جای‌گیری P و سپس ادغام دو خوشه وجود دارد.

سوال ۳

در این سوال به اعمال الگوریتم‌های خوشه بندی Agglomorative با دو روش Single-link و Complete-link روی داده‌های موجود می‌پردازیم. هر دو روش ذکر شده روش‌هایی سلسله مراتبی و از پایین به بالا هستند که با هر کدام از نقاط به عنوان یک خوشه شروع می‌کنند و در هر مرحله به کمک معیار شباهت تعریف شده برای هر کدام از آن‌ها، نزدیک‌ترین خوشه‌ها را با یکدیگر ادغام می‌کنند تا در نهایت به یک خوشه واحد که شامل تمامی داده‌ها است برسند.

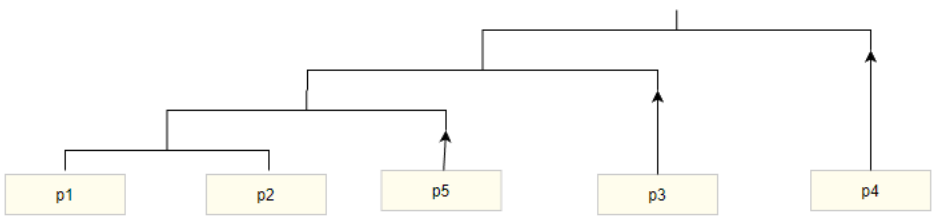
مجموعه داده مورد نظر برای اعمال الگوریتم‌های فوق شامل ۵ نقطه است و فاصله هر زوج نقطه در جدول زیر ذکر شده است:

	P1	P2	P3	P4	P5
P1	1	0.1	0.41	0.55	0.35
P2	0.1	1	0.64	0.47	0.98
P3	0.41	0.64	1	0.44	0.85
P4	0.55	0.47	0.44	1	0.76
P5	0.35	0.98	0.85	0.76	1

روش ۱: Single Link

در این روش معیار شباهت خوشه‌ها، فاصله نزدیک‌ترین نقاط دو خوشه به یکدیگر است. با توجه به تعریف این روش می‌توان در هر مرحله کمترین مقدار فواصل را از جدول فوق انتخاب کرد و خوشه‌های شامل آن‌ها را با یکدیگر ترکیب کرد. در ادامه نحوه عملکرد به صورت مرحله به مرحله روی داده‌های فوق توضیح داده می‌شود.

- **مرحله ۰**: در ابتدا هر نقطه در یک خوشه جدا قرار می‌گیرد. بدین ترتیب نقاط P1 تا P5 به ترتیب در خوشه‌های C1 تا C5 قرار می‌گیرند و در مجموع ۵ خوشه داریم.
- **مرحله ۱**: در این مرحله کمترین فاصله بین دو خوشه از خوشه‌های مرحله قبل انتخاب می‌شود و خوشه‌های آن‌ها با یکدیگر ترکیب می‌شوند. کمترین فاصله بین خوشه متعلق به خوشه‌های C1 و C2 و برابر با ۰.۱ است بنابراین نقاط موجود در این خوشه‌ها در یک خوشه با نام C1,2 قرار می‌گیرند. و خوشه‌ها در پایان این مرحله به شکل {C1,2={P1, P2}، {C3={P3}، {C4={P4} و {C5={P5} درمی‌آیند.
- **مرحله ۲**: کمترین فاصله بعدی بین خوشه‌های مرحله قبل بین C1,2 و C5 و برابر با ۰.۳۵(بین نقاط P1 و P5) است. بنابراین خوشه {C1,2,5={P1, P2, P5} ایجاد می‌شود و خوشه‌های C4 و C3 مشابه با قسمت قبل باقی می‌مانند.
- **مرحله ۳**: دو خوشه بعدی که با یکدیگر ترکیب می‌شوند C1,2,5 و C3 با شباهت ۰.۴۱(بین نقاط P1 و P3) هستند و خوشه {C1,2,5,3={P1, P2, P5, P3} ایجاد می‌شود و خوشه C4 مشابه با مراحل قبل است.
- **مرحله ۴**: در نهایت ۲ خوشه باقی‌مانده با شباهت ۰.۴۴(بین نقاط P3 و P4) با یکدیگر ترکیب می‌شوند و خوشه C1,2,5,3,4={ P1, P2, P5, P3, P4} ایجاد می‌شود.



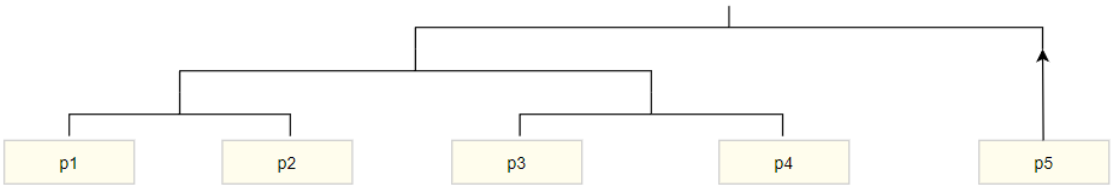
پس از انجام مراحل فوق نمودار Dendrogram به شکل بالا به دست می‌آید. با توجه به شکل فوق، با توقف الگوریتم در هر یک از مراحل می‌توان به تعداد متفاوتی از خوشه‌ها دست یافت که زمان این توقف با توجه به مسئله تعریف می‌شود و نمی‌توان زمان یکسانی را برای همه دیتاست‌ها مشخص کرد.

روش ۲: Complete Link

در این روش معیار شباهت دو خوشه فاصله دورترین نقاط در خوشه‌های متفاوت با یکدیگر است. در ادامه نحوه عملکرد این روش بر روی داده‌های موجود در جدول ابتدای سوال با تفکیک مراحل مشخص شده است.

- **مرحله ۰**: همانند روش Single-link، در این روش نیز با خوشه‌های C1 تا C5 که هر کدام شامل یکی از نقاط P1 تا P5 هستند کار را آغاز می‌کنیم.
- **مرحله ۱**: کمترین فاصله بین ۵ خوشه‌ی موجود از مرحله قبل مربوط به خوشه‌های C1 و C2 است که خوشه C1,2={P1, P2} ایجاد می‌شود و سایر خوشه‌های بدون تغییر باقی می‌مانند.
- **مرحله ۲**: پس از ایجاد خوشه C1,2 فاصله آن با سایر خوشه‌های C3، C4 و C5 برابر با ۰.۶۴(بین P2 و P3)، ۰.۵۵(بین P1 و P4) و ۰.۹۸(بین P2 و P5) محاسبه می‌شود و سایر فاصله‌ها در جدول قابل مشاهده هستند. در این مرحله کمترین فاصله ممکن بین خوشه‌های C3 و C4 و برابر ۰.۴۴ محاسبه می‌شود بنابراین خوشه C3,4={P3, P4} ایجاد می‌شود و خوشه‌های C1,2 و C5 مشابه مرحله قبل هستند.
- **مرحله ۳**: در ابتدای این مرحله سه خوشه C1,2، C3,4 و C5 وجود دارند که فواصل خوشه C1,2 و C5 برابر ۰.۹۸(بین P2 و P5)، بین C3,4 و C5 برابر ۰.۸۵(بین P3 و P5) و بین C1,2 و C3,4 برابر ۰.۶۴(بین P2 و P3) است بنابراین خوشه C1,2,3,4={P1, P2, P3, P4} ایجاد می‌شود و خوشه C5 بدون تغییر باقی می‌ماند.
- **مرحله ۴**: در آخرین مرحله دو خوشه باقی مانده با فاصله ۰.۹۸(بین P2 و P5) با یکدیگر ترکیب می‌شوند و خوشه نهایی C1,2,3,4,5={ P1, P2, P3, P4, P5} را ایجاد می‌کنند که شامل تمامی نقاط موجود در دیتاست است.

پس از انجام مراحل فوق نمودار Dendrogram به شکل زیر به دست می‌آید:



همانند روش قبل در هر نقطه از این روش نیز می‌توان الگوریتم را متوقف کرد و به تعداد خوشه‌های مناسب برای مسئله خود دست یافت.

تمرین عملی

در این تمرین به پاکسازی دیتاست داده شده و سپس بررسی عملکرد هر الگوریتم با پارامترهای متفاوت به کمک ضریب سیلوئت می‌پردازیم.

سوال ۱

در سوال یک به پاکسازی داده‌های موجود می‌نماییم. این پاکسازی شامل مراحل مختلفی از جمله تشخیص و برخورد با سطرهای خالی، تبدیل ویژگی‌های غیر عددی به عددی و نرمال کردن داده‌ها می‌شود که در ادامه به بررسی آن‌ها می‌پردازیم.

تشخیص سطرهای خالی

با بررسی دیتاست موجود می‌توان متوجه شد که ویژگی‌های زیادی نیستند که ناقص هستند و شامل سطرهای خالی می‌شوند اما این ویژگی‌ها دارای وجه اشتراک داشتن مقدار "؟" یا رشته "None" هستند. برای راحتی در برخورد با این مقادیر و سادگی در مشاهده آن‌ها، به کمک دستورات زیر این مقادیر را تبدیل به nan می‌کنیم.

```
df = df.replace('?', np.nan)
df = df.replace('None', np.nan)
```

همچنین با توابعی که به لطف مقدار nan می‌توان تعریف کرد امکان یافتن تعداد سطرهای خالی برای هر ویژگی وجود دارد. این توابع به شکل زیر هستند:

```
def CountMissingCol(df, col):
    return list(df[col].isnull()).count(True)

def CountMissingData(df):
    d = dict()
    for col in df.columns:
        d[col] = CountMissingCol(df, col)
    return d
```

در خروجی این توابع می‌توان مشاهده کرد که تعداد ۹ ویژگی از ویژگی‌های موجود دارای سطر خالی هستند و باید با استراتژی مناسب پر شوند. این ویژگی‌ها به همراه سطرهای خالی به شرح زیر می‌باشند:

Column	Empty spaces
A1Cresult	84748
diag_1	21
diag_2	358
diag_3	1423
max_glu_serum	96420
medical_specialty	49949
payer_code	40256
race	2273
weight	98569

برخورد با سطرهای خالی

برای انجام خوشه‌بندی و دریافت نتایج مناسب، نیاز داریم تا ویژگی‌هایی که دارای نقص در مقادیر خود هستند را بررسی و ایرادزدایی کنیم. از بین ویژگی‌های فوق ویژگی‌هایی که بیش از ۸۰۰۰۰ سطر خالی دارند از مجموعه داده حذف شده‌اند و باقی ویژگی‌ها به وسیله متد مناسب تکمیل شده‌اند. روش انتخاب شده برای پر کردن سطرهای خالی استفاده از تابع fillna از کتابخانه pandas است. همچنین سطرهای دیگری مانند citoglipton و examide که دارای فرمت مناسبی نبودند نیز از مجموعه داده حذف شدند.

تبدیل ویژگی‌های غیر عددی به عددی

بسیاری از الگوریتم‌های یادگیری ماشین و خوشه‌بندی بر روی داده‌های عددی کار می‌کنند و امکان اجرا بر روی داده‌های غیر عددی را ندارند. برای عددی کردن تمام ویژگی‌ها، یک مجموعه با اعضای یکتا از مقادیر ویژگی‌های غیر عددی ایجاد می‌کنیم و هر مقدار را با اندیس خود در مجموعه ایجاد شده جایگزین می‌کنیم. این کار به کمک قطعه کد زیر انجام شده است:

```
values = list(set(df[col]))
df[col] = df[col].map(lambda x: values.index(x))
```

نرمال سازی مقادیر

مقادیر تمام ویژگی‌ها که اکنون عددی هستند را می‌توان به کمک روشی مانند Z-score یا MinMax نرمال کرد تا تاثیر ویژگی‌هایی با بازه مقادیر بزرگ بیشتر از بقیه نباشد. در این تمرین از روش Z-score با فرمول $Zscore = \frac{x-\mu}{\sigma}$ استفاده شده است و به شکل زیر پیاده‌سازی شده است:

```
norm_df = (df-np.mean(df))/np.std(df)
```

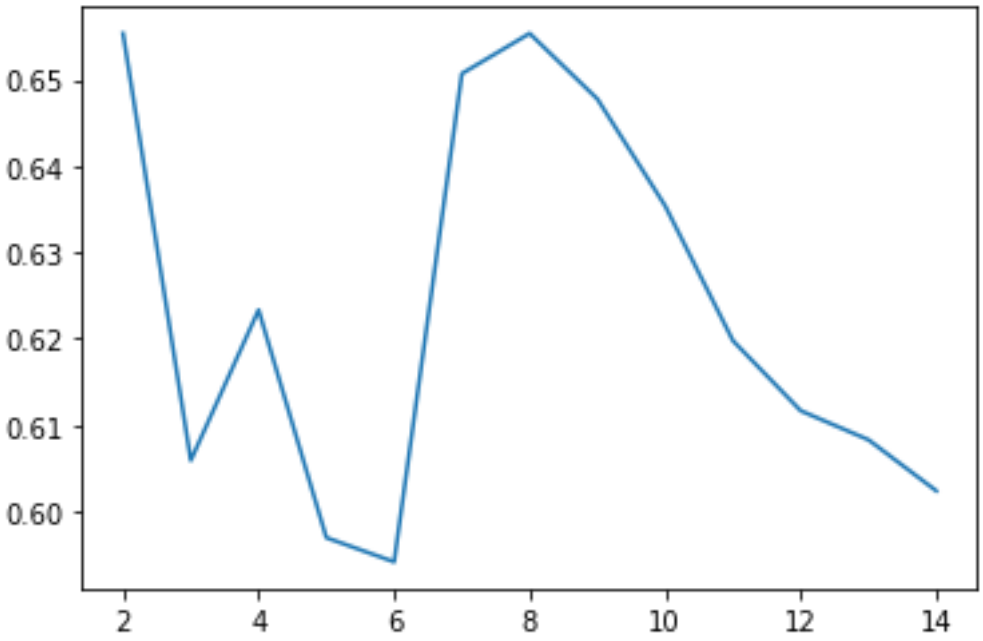
متاسفانه این روش باعث کاهش کیفیت روش‌های استفاده شده در مرحله بعد شد و در نتیجه از اعمال نرمال‌سازی بر روی دیتاست صرف نظر کردیم.

سوال ۲

در این سوال مدل‌های DBSCAN و K-means را با پارامترهای متفاوت آموزش می‌دهیم و ارزیابی می‌کنیم و بهترین مقادیر پارامترها را برای هر کدام از مدل‌ها انتخاب می‌کنیم.

K-means

برای مدل K-means تعداد خوشه‌ها را بین ۲ و ۱۲ قرار می‌دهیم و مقدار ضریب سیلوئت را برای هر یک از حالات محاسبه می‌کنیم. نمودار ضریب سیلوئت نسبت به تعداد خوشه‌ها به شکل زیر می‌باشد:



همانطور که مشاهده می‌شود به ازای تعداد خوشه‌های ۲ و ۸ بهترین مقادیر ضریب سیلوئت محاسبه شده‌اند. برچسب گذاری داده با تعداد ۲ خوشه در فایل ضمیمه قابل مشاهده است.

ایجاد، آموزش و ارزیابی مدل‌های ایجاد شده با دستورات زیر انجام می‌شود:

```
km = KMeans(n_clusters=2)
```

```
km.fit_predict(df.drop('encounter_id', axis=1))
score = silhouette_score(df.drop('encounter_id', axis=1), km.labels_, m
etric='euclidean')
```

در دستورات فوق مشخص است که ستون encounter_id به عنوان ویژگی به الگوریتم داده نشده است چرا که این ستون ویژگی محسوب نشده و یک شناسه یکتا برای نمونه‌های موجود در دیتاست است.