به نام خدا دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) دانشکده مهندسی کامپیوتر



گزارش تمرین دوم درس شبکههای پیچیده

استاد درس: دكتر مصطفى حقير چهرقانى

دانشجو: فاطمه غلامزاده ۹۹۱۳۱۰۰۳

نيم سال اول ۱۴۰۱–۱۴۰۰

سوال اول

پاسخ این سوال با نام Question1.pdf ضمیمه شده است.

سوال دوم

مراحل خوشهبندی به این صورت است:

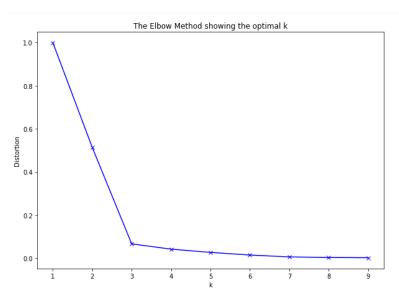
• ابتدا ماتریس لاپلاسین را برای گراف مربوطه حساب میکنیم. برای محاسبه ماتریس لاپلاسین از رابطه زیر استفاده می شود:

L = A - D •

که در آن A ماتریس مجاورت گراف و D یک ماتریس قطری است که هر عنصر روی قطر درجه نود متناظر با آن را نشان میدهد.

ماتریس لاپلاسین استخراج شده به این شکل است:

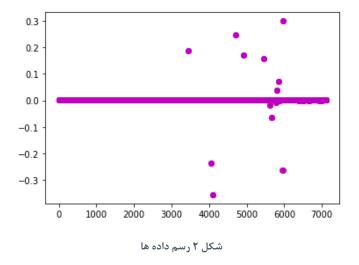
- بعد از محاسبه ماتریس لاپلاسین، مقادیر و بردارهای ویژه را برای آن محاسبه کرده و مرتب می کنیم.
 دومین کوچکترین بردار ویژه را استخراج کرده و از آن به عنوان بردار ویژگی برای خوشهبندی استفاده می کنیم. به این صورت که هر عنصر بردار lambda2 (بردار ویژه ی متناظر با دومین کوچکترین مقدار ویژه) متناظر با یکی از نودهای گراف است و ما قصد داریم که این دادههای یک بعدی را با الگوریتم -k
 شویژه شویندی کنیم.
- برای یافتن تعداد مناسب خوشهها از روش elbow در الگوریتم k-means استفاده شد. نمودار روش elbow در شکل ۱ رسم شده است:



شکل ۱ روش elbow برای یافتن تعداد بهینه ی خوشه ها

همانطور که مشاهده می شود تعداد بهینه برای خوشهها ۳ عدد می باشد زیرا تا ۳ خوشه کاهش مقدار k-means خیلی زیاد بوده اما از آن به بعد این کاهش محسوس نیست. بنابراین تعداد ۳ خوشه را برای الگوریتم انتخاب می کنیم.

یک روش دیگر برای اطمینان از تعداد مناسب خوشهها رسم این دادههای یک بعدی است. نمودار دادهها در شکل ۲ نشان داده شده است.



با توجه به شکل تعدادی از دادهها مقادیر بزرگتر از صفر، تعداد مقادیر کوچکتر از صفر و تعداد زیادی مقادیر خیلی نزدیک به صفر دارند. بنابراین از نمودار دادهها نیز مشخص است که تعداد ۳ خوشه می تواند تعداد مناسبی برای دادهها باشد.

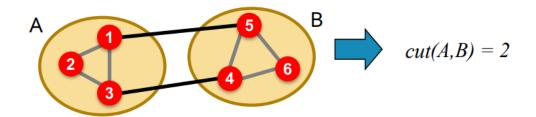
فایل idx_Q2.txt در فولدر پاسخها ضمیمه شده است. نودهایی که داخل ۲ تا از خوشهها قرار گرفتهاند در زیر آورده شده است و سایر نودها در خوشه سوم قرار دارند.

Cluster1: 3194, 3195, 3244, 3245, 6258, 6259

Cluster2: 2304, 2305, 4167, 4168, 4540, 4541, 5413, 5414, 6266, 6267

محاسبه min Cut:

برای محاسبه این معیار باید تعداد یالهایی که یک سرشان در یک کلاستر و سر دیگرشان در کلاستر دیگری است را بشماریم. به عنوان مثال معیار cut در شکل زیر برابر ۲ است:



هر چقدر معیار cut کمتر باشد کیفیت خوشهبندی مناسبتر است.

مقدار cut پس از خوشهبندی گراف داده شده در سوال: <mark>0</mark>

با توجه به اینکه معیار cut مقدار کمی به دست آمده به نظر میرسد که کیفیت کلاسترینگ مناسب است.

محاسبه modularity:

برای محاسبه ماژولاریتی یک تابع نوشته شده که رابطه زیر را پیاده می کند:

$$\frac{1}{2m}\sum_{s\in S}\sum_{i\in s}\sum_{j\in s}\left(A_{ij}-\frac{k_ik_j}{2m}\right)$$

مقدار ماژولاریتی به دست آمده: 0.0001587803677129725

مقدار گرده شده ماژولاریتی: <mark>0.0001</mark>

سوال سوم

برای انجام این سوال یک تابع بازگشتی نوشته شده است که در ابتدا ماتریس ماژولاریتی را برای گراف ورودی محاسبه می کند. برای محاسبه ماتریس ماژولاریتی از رابطه زیر استفاده می کنیم:

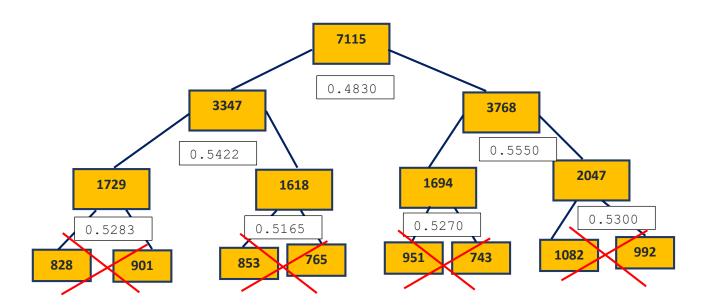
• Modularity matrix: $B_{ij} = A_{ij} - \frac{k_i k_j}{2m}$

سپس این تابع بزرگترین بردار ویژه ماتریس ماژولاریتی را مییابد و با استفاده از علامتهای عناصر آن گراف به دو بخش تقسیم میشود. سپس برای خوشههایی که تا این مرحله داریم مقدار modularity میشود و با مقدار قبلی آن مقایسه میشود. اگر کمتر باشه return میکند و اگر بیشتر باشد تابع به صورت بازگشتی روی دو زیرگراف حاصل صدا زده میشود.

مقادیر ماژولاریتی که این تابع به آنها دست یافته به ترتیب در زیر آورده شده است:

$$0.4830 - 0.5422 - 0.5283 - 0.5165 - 0.5550 - 0.5270 - 0.5300$$

درخت بازگشتی که الگوریتم در آن طی میشود به صورت زیر است. اعداد داخل مستطیلهای زردرنگ نشان دهنده تعداد نودها هستند و اعداد داخل مستطیلهای سفیدرنگ مقدار modularity در آن مرحله از تقسیم را نشان میدهد.



بنابراین ابتدا ۷۱۱۵ نود در دو دستهی ۳۳۴۷ و ۳۷۶۸ تایی قرار می گیرند و مقدار ماژولاریتی ۴۸،۰ به دست می آید. سپس دستهی ۳۷۶۸ تایی به دو دسته تقسیم می شود با اندازههای ۲۰۴۷ و ۱۶۹۴ و مقدار ماژولاریتی می آید. سپس دستهی که ماژولاریتی صعودی است تقسیم بندی ادامه می یابد و هر کدام از دستههای ۲۰۴۷ تایی و

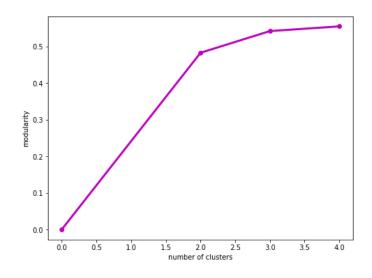
۱۶۹۴ تایی هم به دو خوشه تقسیم میشوند اما مقدار ماژولاریتی در هر دوی این موارد کاهشی است پس این شاخهها prune میشوند.

بعد از آن از شاخهی سمت چپی کار ادامه پیدا می کند و روند دقیقا مانند توضیحاتی است که برای شاخه سمت راست ذکر شد. بنابراین در آخر تعداد خوشهها ۴ عدد است. جدول زیر تعداد نودهای هر خوشه را نشان می دهد.

تعداد نودهای درون خوشه	شماره خوشه
1779	1
1811	٢
1894	٣
7.47	۴

فایل idx_Q3.txt در فولدر پاسخ ضمیمه شده است.

نمودار ماژولاریتی برحسب تعداد خوشهها:



سوال چهارم

ده گرهای که بیشتری امتیاز authority را دارند به همراه امتیاز آنها در تصویر زیر آورده شده است. عدد سمت چپ شماره نود و عدد سمت راست امتیاز authority را برای آن نود مشخص می کند.

```
('2398', 1.0)

('4037', 0.9977207958136023)

('3352', 0.9024990712420015)

('1549', 0.8933904532987473)

('762', 0.8752334923138549)

('1297', 0.8736172836899779)

('3089', 0.8734033014392443)

('2565', 0.8620015229598714)

('15', 0.8536687098722228)

('2625', 0.8530062804275313)
```

ده گرهای که بیشتری امتیاز hub را دارند به همراه امتیاز آنها آورده شده است. عدد سمت چپ شماره نود و عدد سمت راست امتیاز hub را برای آن نود مشخص می کند.

```
('2565', 1.0)

('766', 0.9539738367956294)

('2688', 0.8109012307736835)

('457', 0.8083998263718667)

('1166', 0.7572151832677456)

('1549', 0.7208933608812452)

('11', 0.6205408521474032)

('1151', 0.5761852013414268)

('1374', 0.5628874083453793)

('1133', 0.493479413970747)
```

سوال پنجم

مقدارهای precision و recall برای هر یک از روشها در جدول زیر آورده شده است:

recall	precision	معيار
0.2014	0.5665	Common_neighbors
0.1600	0.5017	jaccard
0.2872	0.6926	Adamic Adar
0.08	0.15	Shortest path

همانطور که مشاهده می شود بهترین عملکرد را معیار Adamic_Adar و بدترین عملکرد را معیار shortest دارد. اگر بخواهیم به ترتیب عملکرد معیارها را مرتب کنیم:

Shortest path< jaccard < Common_neighbor < Adamic_Adar