

**گزارش‌های تمرین دوم جبر خطی**

سید محمد سروش سجادی

فروردین ۱۴۰۲



دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

**سوال ۴:**

در این سوال از ماژول numpy برای انجام محاسبات جبری و برداری و opencv برای کار با ورودی که عکس می‌باشند استفاده شده است. در ابتدا کلاسی به نام K\_Means را مشاهده می‌نمایید که در آن عملکرد الگوریتم پیاده‌سازی شده است. ویژگی‌های این کلاس عبارتند از:

- Num\_center: تعداد خوشه‌ها

- data\_dimension: بعد عکس‌های ورودی

- data: داده‌های ورودی که به صورت برداری از بردارها می‌باشد

- z: برداری از نماینده‌های هر خوشه

- c: بردار انتساب داده‌ها به خوشه‌ها

- termination: نماینگر شرایط خاتمه الگوریتم

- G: دیکشنری یا مپی از بردار انتساب هر خوشه که میان داده‌ها و نماینده‌ آن‌ها ارتباطی ایجاد می‌کند برای نمایش همسایگی‌ها

در تابع سازنده این کلاس، تعداد خوشه‌ها، بعد عکس‌ها و داده‌های ورودی به صورت برداری از بردارها به کلاس وارد می‌شوند و مقداردهی اولیه انجام می‌گردد.

متود dist وظیفه محاسبه‌ی فاصله‌ی اقلیدسی میان دو نقطه یا در واقع دو داده را دارد که به دلیل برداری بودن داده‌ها وجود [0] قبل از [i] در پیاده‌سازی لازم می‌باشد.

متود ranmdom\_init\_center وظیفه مقداردهی اولیه به نمایندگان خوشه‌ها را دارد که این بردارهای تصادفی در لیست z ذخیره می‌شوند.

متود clustering بر اساس فواصل داده‌ها تا نمایندگان، آن‌ها را که نزدیک‌ترین نماینده انتساب می‌دهد. همچنین مقادیر c وG را نیز بروزرسانی می‌نماید.

متود Update\_Centers وظیفه میانگین‌گیری از داده‌های هر خوشه را بر عهده دارد و سپس این میانگین برابر با نماینده جدید خوشه می‌شود. همچنین بررسی می‌شود که آیا نماینده‌ای تغییر پیدا کرده است یا خیر و مقدار termination را تنظیم می‌نماید.

متود Show\_Samples بر اساس داده‌های ذخیره شده در c، ۲۰ نمونه اول داده‌های ورودی را برای هر خوشه دخیره می‌نماید.

متود Show\_Samples\_KNN با داشتن یک ورودی مثل تعداد همسایه مورد نظر، نزدیک‌ترین همسایه‌ها به هر خوشه را ذخیره می‌نماید.

متود Do بدون هیچ پارامتری خوشه بندی را انجام داده و سپس ۲۰ نمیانده از هر خوشه را ذخیره می‌نمیاد. اما متود Do\_KNN پارامتر تعداد همسایه را دریافت کرده و همسایگی مورد نظر از هر خوشه را ذخیره می‌نمیاد.

لازم به ذکر است که در حال اجرا الگوریتم تعداد دفعات تکرار اجرا را تا خاتمه چاپ می‌نماید. فایل لاگ اجراهای زیر نیز ضمیمه شده است.

در ادامه می‌توانید خروجی ۲۰ نمونه هر خوشه را بر اساس تعداد خوشه‌ها مشاهده نمایید:

k = 3 :

نماینده خوشه اول:

داده‌ها: 

نماینده خوشه دوم:

داده‌ها: 

نماینده خوشه سوم:

داده‌ها: 

k = 4 :

نماینده خوشه اول:

داده‌ها: 

نماینده خوشه دوم:

داده‌ها: 

نماینده خوشه سوم:

داده‌ها: 

نماینده خوشه چهارم:

داده‌ها: 

k = 5 :

نماینده خوشه اول: 

داده‌ها: 

نماینده خوشه دوم: 

داده‌ها: 

نماینده خوشه سوم:

داده‌ها: 

نماینده خوشه چهارم: 

داده‌ها: 

نماینده خوشه پنجم:

داده‌ها: 

k = 6 :

نماینده خوشه اول: 

داده‌ها: 

نماینده خوشه دوم: 

داده‌ها: 

نماینده خوشه سوم: 

داده‌ها: 

نماینده خوشه چهارم: 

داده‌ها: 

نماینده خوشه پنجم: 

داده‌ها: 

نماینده خوشه ششم: 

داده‌ها: 

k = 7 :

نماینده خوشه اول: 

داده‌ها: 

نماینده خوشه دوم: 

داده‌ها: 

نماینده خوشه سوم: 

داده‌ها: 

نماینده خوشه چهارم: 

داده‌ها: 

نماینده خوشه پنجم: 

داده‌ها: 

نماینده خوشه ششم: 

داده‌ها: 

نماینده خوشه هفتم: 

داده‌ها: 

**سوال ۵:**

برای این سوال از همان کد سوال قبل استفاده شده است و تنها تفاوت در نحوه خواندن ورودی و خروجی است. خروجی این سوال چاپ می‌گردد. در ادامه خروجی یک نمونه اجرا را مشاهده می‌نمایید.

خروجی این اجرا در فایل لاگ نیز قابل مشاهده است.

در این الگوریتم id مشتریان و جنسیت آنها در خوشه بندی اثر داده نشده است. می‌توان برای جنسیت ضریبی متناسب با سایر داده‌ها قرار داد و آن را نیز دخیل کرد اما در نهایت مقدار آن برای نمایندگان مشخص کننده جنسیت خاصی نخواهد بود وتنها نسبت جنسیت‌ها را نمایان می‌کند.

--------------------------------- The First Run ----------------------------------

----------------- 0 -----------------

----------------- 1 -----------------

----------------- 2 -----------------

----------------- 3 -----------------

----------------- 4 -----------------

----------------- 5 -----------------

----------------- 6 -----------------

----------------- 7 -----------------

----------------- 8 -----------------

----------------- 9 -----------------

----------------- 10 -----------------

----------------- 11 -----------------

1 Cluster:

The center is: [ 0.46153846 32.69230769 86.53846154 82.12820513]

1 [176 0 30 88 86]

2 [162 0 29 79 83]

3 [166 0 36 85 75]

4 [172 1 28 87 75]

5 [158 0 30 78 78]

2 Cluster:

The center is: [ 0.425 43.15 55.0875 49.4 ]

1 [78 1 40 54 48]

2 [77 0 45 54 53]

3 [84 0 46 54 44]

4 [86 1 48 54 46]

5 [97 0 47 60 47]

3 Cluster:

The center is: [ 0.52777778 40.66666667 87.75 17.58333333]

1 [167 1 42 86 20]

2 [171 1 40 87 13]

3 [173 1 36 87 10]

4 [151 1 43 78 17]

5 [153 0 44 78 20]

4 Cluster:

The center is: [ 0.36363636 45.09090909 25.72727273 20.22727273]

1 [37 0 42 34 17]

2 [35 0 49 33 14]

3 [25 0 54 28 14]

4 [19 1 52 23 29]

5 [15 1 37 20 13]

5 Cluster:

The center is: [ 0.39130435 25.52173913 26.30434783 78.56521739]

1 [22 1 25 24 73]

2 [26 1 29 28 82]

3 [14 0 24 20 77]

4 [16 1 22 20 79]

5 [32 0 21 30 73]

--------------------------------- The Second Run ---------------------------------

----------------- 0 -----------------

----------------- 1 -----------------

----------------- 2 -----------------

----------------- 3 -----------------

----------------- 4 -----------------

----------------- 5 -----------------

----------------- 6 -----------------

1 Cluster:

The center is: [ 0.35849057 33.39622642 58.05660377 48.77358491]

1 [95 0 32 60 42]

2 [98 0 27 60 50]

3 [78 1 40 54 48]

4 [82 1 38 54 55]

5 [113 0 38 64 42]

2 Cluster:

The center is: [ 0.55882353 41.64705882 88.73529412 16.76470588]

1 [171 1 40 87 13]

2 [167 1 42 86 20]

3 [173 1 36 87 10]

4 [151 1 43 78 17]

5 [153 0 44 78 20]

3 Cluster:

The center is: [ 0.41666667 25.25 25.83333333 76.91666667]

1 [22 1 25 24 73]

2 [14 0 24 20 77]

3 [16 1 22 20 79]

4 [24 1 31 25 73]

5 [32 0 21 30 73]

4 Cluster:

The center is: [ 0.44 54.06 40.46 36.72]

1 [43 1 48 39 36]

2 [56 1 47 43 41]

3 [45 0 49 39 28]

4 [55 0 50 43 45]

5 [41 0 65 38 35]

5 Cluster:

The center is: [ 0.46153846 32.69230769 86.53846154 82.12820513]

1 [176 0 30 88 86]

2 [162 0 29 79 83]

3 [166 0 36 85 75]

4 [172 1 28 87 75]

5 [158 0 30 78 78]

--------------------------------- The Third Run ---------------------------------

----------------- 0 -----------------

----------------- 1 -----------------

----------------- 2 -----------------

----------------- 3 -----------------

----------------- 4 -----------------

----------------- 5 -----------------

----------------- 6 -----------------

----------------- 7 -----------------

1 Cluster:

The center is: [ 0.46153846 32.69230769 86.53846154 82.12820513]

1 [176 0 30 88 86]

2 [162 0 29 79 83]

3 [166 0 36 85 75]

4 [172 1 28 87 75]

5 [158 0 30 78 78]

2 Cluster:

The center is: [ 0.39130435 45.2173913 26.30434783 20.91304348]

1 [37 0 42 34 17]

2 [35 0 49 33 14]

3 [25 0 54 28 14]

4 [19 1 52 23 29]

5 [27 0 45 28 32]

3 Cluster:

The center is: [ 0.52777778 40.66666667 87.75 17.58333333]

1 [167 1 42 86 20]

2 [171 1 40 87 13]

3 [173 1 36 87 10]

4 [151 1 43 78 17]

5 [153 0 44 78 20]

4 Cluster:

The center is: [ 0.41772152 43.08860759 55.29113924 49.56962025]

1 [78 1 40 54 48]

2 [77 0 45 54 53]

3 [84 0 46 54 44]

4 [86 1 48 54 46]

5 [97 0 47 60 47]

5 Cluster:

The center is: [ 0.39130435 25.52173913 26.30434783 78.56521739]

1 [22 1 25 24 73]

2 [26 1 29 28 82]

3 [14 0 24 20 77]

4 [16 1 22 20 79]

5 [32 0 21 30 73]

**سوال ۶:**

در این سوال الگوریتم گرام اشمیت پیاده‌سازی شده است. کلاس گرام اشمیت دارای ویژگی‌های زیر می‌باشد:

- q: لیست پایه‌های یکه متعامد

- qq: معادل کیو تیلدا در الگورتم گرام اشمیت می‌باشد

- data: بردارهای ورودی است که روی کره آر۵ قرار دارند

- dimension: بعد داده‌های ورودی است که در اینجا ۵ است

- size: اندازه داده ورودی می باشد

- base\_size: خام اندازه ورودی است که در ابتدا برابر با همان سایز است

- V: لیستی از بردارهایی است که مستقل خطی می‌باشند

- linear\_independece: نمایانگر استقلال خطی کل بردارهای ورودی است

در تابع سازنده ویژگی‌های بالا مقداردهی اولیه می‌شوند.

در متود orthogonalization عملیات محاسبه‌ی کیو تیلدا که یک عمود است انجام می‌شود.

متود test بررسی می‌کند که کیو تیلدا بدست آمده صفر نباشد (در این پیاده سازی با توجه به اینکه اعداد در رنج بی‌نهایت است و سخت افزار دارای دقتی است به جای بررسی صفر بودن کیو تیلدا، بسیار کوچک بودن آن بررسی می‌شود).

متود normalization کیو تیلدا را نرمال‌سازی می‌کند تا نرم آن برابر ۱ شود.

متود show-result خروجی الگوریتم را نمایش می‌دهد که پاسخ سوالات بیان شده است.

متود Do این الگوریتم را اجرا می‌نماید.

در ادامه می‌توانید خروجی این الگوریتم را برای n های خواسته شده مشاهده نمایید، این خروجی در فایل لاگ نیز ضمیمه شده است.

---------------------- n = 2 ----------------------

These 2 vectors are linear independed!

V = all vectors and is as follows:

[0.40332238 0.47955405 0.12862588 0.57454929 0.51059521]

[ 0.05468123 -0.07252834 0.02087548 -0.21517461 0.26179326]

And the orthonormal basis is:

[0.40332238 0.47955405 0.12862588 0.57454929 0.51059521]

[ 0.15558443 -0.20636477 0.05939698 -0.61223599 0.74487996]

---------------------- n = 3 ----------------------

These 3 vectors are linear independed!

V = all vectors and is as follows:

[0.09815347 0.54645467 0.00985986 0.73655268 0.38619442]

[ 0.06933382 -0.23130136 0.18788247 0.16949086 -0.01838733]

[ 0.21980691 -0.0120551 0.06126188 -0.14802932 0.24195098]

And the orthonormal basis is:

[0.09815347 0.54645467 0.00985986 0.73655268 0.38619442]

[ 0.19795725 -0.66039605 0.53642935 0.48391886 -0.05249826]

[ 0.60347725 -0.03309715 0.16819376 -0.40641274 0.66427353]

---------------------- n = 5 ----------------------

These 5 vectors are linear independed!

V = all vectors and is as follows:

[0.21785 0.6244826 0.21613173 0.50075045 0.51487759]

[ 0.00342978 -0.33144569 0.42026518 -0.09576003 0.31726789]

[ 0.12125727 0.07891957 -0.0673309 -0.22684251 0.1018573 ]

[ 0.43560768 -0.11294166 0.05801474 0.10033624 -0.16926213]

[ 0.0202506 -0.10177562 -0.14787851 0.06874309 0.11009145]

And the orthonormal basis is:

[0.21785 0.6244826 0.21613173 0.50075045 0.51487759]

[ 0.00544808 -0.52648928 0.66757577 -0.15211128 0.50396836]

[ 0.41039955 0.26710608 -0.22788381 -0.76775654 0.34473965]

[ 0.88079302 -0.22836655 0.11730504 0.20287857 -0.3422458 ]

[ 0.09103606 -0.45752988 -0.6647843 0.3090329 0.49491348]

---------------------- n = 7 ----------------------

These 7 vectors are linear depended!

V is as follows:

[0.30887097 0.33835505 0.72053419 0.51156854 0.09613894]

[ 0.43320736 -0.05160289 -0.26099725 0.09653877 0.23222815]

[-0.16074333 0.13777439 0.06828061 -0.18090437 0.4824139 ]

[ 0.01479192 0.28661391 -0.10890101 -0.03134543 -0.07326707]

[-0.11636015 0.00336322 -0.08826037 0.18446309 0.04193321]

So the orthonormal basis for V is:

[0.30887097 0.33835505 0.72053419 0.51156854 0.09613894]

[ 0.76378273 -0.09098043 -0.46016114 0.17020635 0.40943868]

[-0.28643401 0.24550486 0.12167153 -0.32235963 0.8596297 ]

[ 0.04664193 0.90375206 -0.34338707 -0.09883854 -0.231026 ]

[-0.48684162 0.01407144 -0.36927438 0.77177888 0.17544523]

---------------------- n = 19 ----------------------

These 19 vectors are linear depended!

V is as follows:

[0.1898824 0.48914344 0.70886305 0.45164294 0.13496369]

[ 0.29020031 0.01977941 -0.03306752 -0.23168754 0.46902604]

[ 0.05528605 0.09517927 0.06767814 -0.19392622 -0.12924433]

[ 0.21662704 0.06723343 -0.15316377 0.1051132 -0.09574394]

[ 0.01997978 -0.03475193 0.01946269 0.00133883 -0.00886303]

So the orthonormal basis for V is:

[0.1898824 0.48914344 0.70886305 0.45164294 0.13496369]

[ 0.48409401 0.03299477 -0.05516118 -0.38648666 0.78239992]

[ 0.20747398 0.35718271 0.25397822 -0.72775396 -0.4850199 ]

[ 0.70237847 0.21799362 -0.49660896 0.3408127 -0.31043437]

[ 0.43956421 -0.76455826 0.42818807 0.02945493 -0.19499066]