آزمایش چهارم (پیاده سازی الگوریتم $k-mean$)

۱.۴ پیش گزارش

۱. مفهوم الگوریتم با نظارت و بدون نظارت را بیان کند.

۲. مزیت های الگوریتم های با نظارت را نسبت به الگوریتم های بدون نظارت و بر عکس بیان کنید

۳. الگوریتم دسته بندی $k-mean$ را با جزییات توضیح دهید.

۴. کاربرد های الگوریتم $k-mean$ را بنویسید.

۲.۴ مقدمه

در آزمایشات گذشته با استفاده از شبکه عصبی چند لایه به دسته بندی داده های با برچسب پرداخته شد.در این آزمایش قصد داریم به دسته بندی داده های بدون برچسب بپردازیم.
به یادگیری بر روی داده های بدون برچسب و تلاش برای پیدا کردن الگوهای نهفته در آن ها یادگیری بدون نظارت می گویند. یکی از پرکاربردترین الگوریتم ها در یادگیری بدون نظارت الگوریتم $k-mean$ است. در ادامه به برررسی روند اجرای الگوریتم $k-mean$ پرداخته می شود.

۱.۲.۴ الگوریتم $k-mean$

الگوریتم $k-mean$ یک الگوریتم بازگشتی است که با یک فرض اولیه درباره مراکز دسته ها آغاز می شود .
در هر مرحله از اجرای الگوریتم مراحل زیر اجرا می شود:

۱. پیدا کردن دسته متناظر با تمام نقاط

۲. تازه سازی مراکز دسته ها

در مرحله اول اجرای الگوریتم فاصله تمام نقاط تا مراکز دسته ها محاسبه می شود و سپس هر داده متعلق به دسته ای که کمترین فاصله با آن را دارد می شود.
بعد از پیدا کردن دسته های متناظر با هر داده ، در مرحله دوم میانگین داده های متعلق به یک دسته به عنوان مرکز دسته در نظر گرفته می شود.
مراحل ۱ و ۲ تا زمانی که مراکز دسته ها تغییر نکند و یا تغییرات خیلی کمی داشته باشه ادامه می یابد.
بعد از ثابت شدن مراکز دسته ها الگوریتم همگرا می شود.
روند کلی اجرای الگوریتم در شکل ۱.۴ نمایش داده شده است

K-MEANS(P, k)

Input: a dataset of points $P = \{p_1, \ldots, p_n\}$, a number of clusters k
Output: centers $\{c_1, \ldots, c_k\}$ implicitly dividing P into k clusters

```
1  choose k initial centers C = {c_1, ..., c_k}
2  while stopping criterion has not been met
3      do ▷ assignment step:
4          for i = 1, ..., N
5              do find closest center c_k ∈ C to instance p_i
6                  assign instance p_i to set C_k
7          ▷ update step:
8          for i = 1, ..., k
9              do set c_i to be the center of mass of all points in C_i
```

شکل ۱.۴: مراحل اجرای الگوریتم $k - mean$

در زمان تست دسته ای که به داده ورودی کمترین فاصله را دارد به داده ورودی نسبت داده می شود.

۳.۴ شرح آزمایش

در این قسمت ابتدا به پیاده سازی الگوریتم $k - mean$ پرداخته می شود . بعد از پیاده سازی الگوریتم به تست و بررسی کاربردها و معایب الگوریتم پرداخته می شود.

۱.۳.۴ پیاده سازی الگوریتم

در شکل ۱.۴ مراحل اجرای الگوریتم $k - mean$ نمایش داده شده است . با توجه به مراحل اجرای الگوریتم به پیاده سازی الگوریتم بپردازید. دقت کنید پیاده سازی باید در محیط پایتون انجام شود و کد پیاده سازی شده باید مستقل از تعداد کلاس و نوع داده ورودی باشد ، به بیانی دیگر کد پیاده سازی شده باید به ازای هر داده ورودی و هر تعداد کلاس کارآیی

داشته باشد(تعداد کلاس ها و داده را از ورودی دریافت کنید.)
برای پیاده سازی الگوریتم مراحل زیر را انجام دهید:

۱. تابعی بنویسید که مراکز دسته ها را با استفاده از اعداد تصادفی مقدار دهی اولیه کند. ورودی این الگوریتم تعداد کلاس ها و خروجی آن مقدار اولیه مراکز دسته ها است.

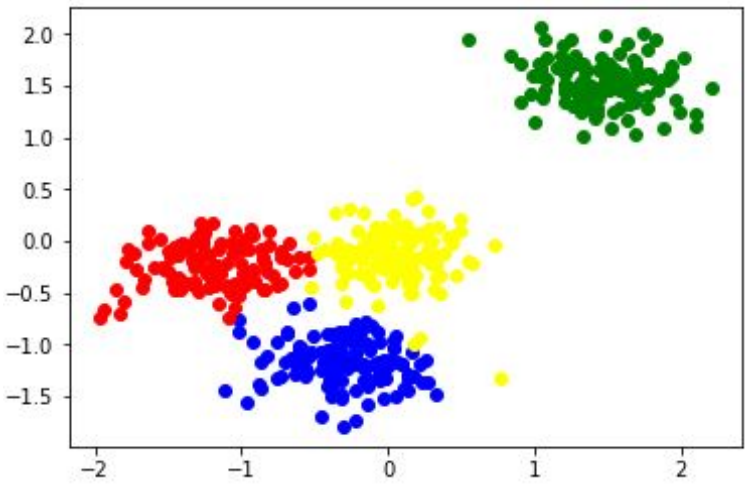
۲. تابعی بنویسید که یک داده و مراکز دسته ها را به عنوان ورودی بگیرد و اندیس نزدیکترین دسته به داده ورودی در آرایه را برگرداند.

۳. تابعی بنویسید که آرایه ای از داده ها را از ورودی دریافت کند و سپس میانگین داده های ورودی را باز گرداند.

۴. با استفاده از توابع پیاده سازی شده الگوریتم $k-mean$ را پیاده سازی کنید

۲.۳.۴ تست الگوریتم

مجموعه ای از داده ها در اختیارتان قرار می گیرد .
الگوریتم را بر روی مجموعه داده هایی که در اختیارتان قرار گرفته است با تعداد دسته های برابر با ۲ و۳ و۴ اجرا کنید و دسته های خروجی را مشاهده کنید. دسته های خروجی را به رنگ های مختلف مانند (شکل۲.۴)رنگ آمیزی کنید.



شکل ۲.۴: دسته بندی انجام شده توسط الگوریتم $k-mean$

۳.۳.۴ ارزیابی الگوریتم

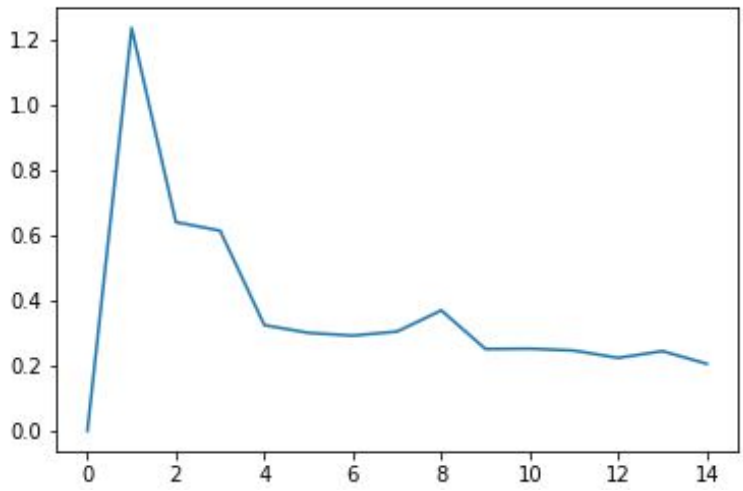
الگوریتم$k-mean$ در دسته الگوریتم های بدون نظارت قرار میگیرد.

تعداد دسته ها و شرایط اولیه برای مراکز دسته ها در عملکرد این الگوریتم تاثیر فراوانی می گذارند برای ارزیابی الگوریتم $k-mean$ مراحل زیر را دنبال کنید:

۱. تابعی بنویسید که نقاط و دسته بندی متناظر با آن ها را به از ورودی بگیرد، سپس برای هر دسته میانگین فاصله نقاط متعلق به دسته مورد نظر را تا مرکز دسته محاسبه کند به این عدد خطای هر دسته گفته می شود.

۲. تابعی بنویسید که نقاط و دسته بندی متناظر با آن ها را به از ورودی بگیرد ومیانگین خطای دسته ها را به عنوان خطای الگوریتم محاسبه کندو به عنوان خروجی باز گرداند.

بعد از پیاده سازی تابع بر روی مجموعه داده هایی که در اختیارتان قرار داده شده است ،الگوریتم$k-mean$ را به ازای دسته های ۱ تا ۱۵ اجرا کنید و در هر مرحله خطای الگوریتم را محاسبه کنید . در انتها نمودار خطا بر حسب تعداد دسته ها را مانند شکل ۳.۴ رسم کنید.



شکل ۳.۴: نمودار خطای خروجی الگوریتم برحسب تعداد دسته (محور افقی تعداد دسته ها و محور عمودی خطا را تشان می دهد.)

۴.۳.۴ محدودیت های $kmean$

در این قسمت به بررسی محدودیت های الگوریتم $k-mean$ پرداخته می شود.

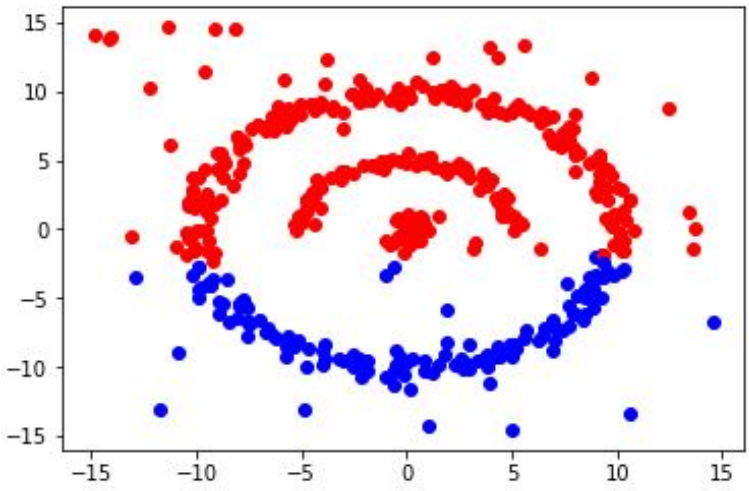
مجموعه ای از داده ها در اختیارتان قرار داده می شود. الگوریتم $k-mean$ پیاده سازی شده را بر روی مجموعه داده ای که در اختیارتان قرار داده شده با تعداد دسته های ۴،۳،۲ اجرا کنید و دسته های مختلف را رنگ آمیزی کنید و نمودار خطا بر حسب تعداد دسته ها را رسم کنید.

در صورت پیاده سازی صحیح الگوریتم خروجی مشابه تصویر ۴.۴ خواهد بود.

۴.۴ کاهش حجم عکس به وسیله $k-mean$ (این بخش شامل نمره مازاد، امتیازی، می باشد)

یکی از کاربرد های الگوریتم $k-mean$ کم کردن حجم عکس است.

یک عکس در اختیارتان قرار داده می شود .در این قسمت قصد داریم با استفاده از الگوریتم $k-mean$ پیاده سازی شده به



شکل ۴.۴: محدودیت های $kmean$(محورها متناضر با ویژگی های داده های ورودی است)

کم کردن حجم عکس بپردازیم.

برای کاهش حجم عکس مراحل زیر را دنبال کنید:

۱. با استفاده از دستور $imread$ $(filename)$ عکس $imagesmall$ را از پوشه مجموعه داده(این پوشه در اختیارتان قرار داده می شود)و ارد محیط پایتون کنید(با بار گذاری عکس متغیر مربوط به آن در محیط پایتون یک آرایه ۳ بعدی با تعداد سطرها و ستون های ۸۰۰ خواهد بود. سطرها و ستون ها متناظر با مکان پیکسل ها خواهد بود و هر بعد آرایه متناظر با مقادیر R و G وB است.).

۲. برای استفاده از الگوریتم $k-mean$ که پیاده سازی کرده اید ابتدا آرایه۳ بعدی عکس را به یک آرایه ۲ بعدی تبدیل کنید درآرایه دو بعدی هر سطر متناظر با یک پیکسل خواهد بود.

۳. الگوریتم $k-mean$ را با تعداد کلاس برابر با ۱۶ اجرا کنید سپس مقدار هر پیکسل را با مقدار مرکز دسته متناظر جایگذاری کنید . عکس خروجی را رسم کنید و با عکس اولیه مقایسه کنید

۵.۴ تمرین

۱. با توجه به نمودار خطا بر حسب تعداد کلاس ها که در قسمت ارزیابی الگوریتم رسم کرده اید حالت بهینه الگوریتم در چه تعداد کلاس رخ می دهد؟

۲. به نظر شما چرا الگوریتم توانایی دسته بندی صحیح داده ها را در قسمت محدودیت های $k-mean$نداشت؟ برای بهبود الگوریتم چه پیشنهادی دارید؟

۳. الگوریتم $k-mean$ پیاده سازی شده را در محیط متلب مجددا پیاده سازی کنید.

۴. دسته بندی داده های قسمت اول آزمایش را با استفاده از دستور $kmean$ در متلب (دستور آماده متلب) تکرار کنید.

۵. فرض کنید شمادارای یک فروشگاه سوپر مارکت هستید و از طریق کارت های عضویت ، اطلاعات اصلی در مورد مشتریان خود مانند شناسه مشتری ، سن ، جنسیت ، درآمد سالانه آن ها در اختیار دارید. نمره خرید چیزی است که شما بر اساس پارامترهای تعریف شده مانند رفتار مشتری و خرید به مشتری اختصاص می دهید. با استفاده از $k-mean$ شبکه ای را طراحی کنید تا مشتری هایی را که راحت تر می توان ترغیب به خرید کرد شناسایی کنید.
داده این مساله را می توانید از $https://www.kaggle.com/vjchoudhary7/customer-segmentation-tutorial-in-python$ دانلود کنید.