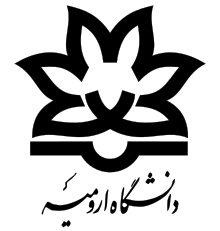
به نام خدا



**دانشکده کامپیوتر**

**"فشرده سازی تصویر با تقریب رتبه پایین"**

**جبرخطی کاربردی**

**استاد : دکتر صباحی**

**گرداورنده : سلوی بهنام اصل**

**974421008**

**نیمسال اول 1401-1402**

امروزه، تجزیه مقادیر منفرد در بسیاری از شاخه‌های علوم و نجوم کاربرد دارد. این روش، در یادگیری ماشین و آمار توصیفی و مدل‌سازی آماری نیز بسیار مورد استفاده قرار می‌گیرد.

کاربردهای ریاضی SVD شامل محاسبه شبه معکوس، تقریب ماتریس و تعیین رتبه، دامنه و فضای تهی یک ماتریس است. SVD همچنین در کلیه زمینه‌های علمی و مهندسی بسیار مفید است مانند پردازش سیگنال، حداقل مربعات مناسب داده‌ها و کنترل فرایند.

البته که محاسبه SVD برای یک ماتریس دارای پیچیدگیِ زمانی خوبی نیست ولی به هر حال این روش یکی از روش‌های شناخته شده در کاهشِ ویژگی‌ها در داده‌کاوی می‌باشد و کاربرد دیگرِ آن که فشرده سازی داده است در ادامه مورد بحث قرار خواهد گرفت. در زبان‌های برنامه‌نویسی مانند Python و R می‌توان کتابخانه‌های مختلفی را پیدا کرد که این عملیات را به سادگی برای شما انجام می‌دهند.

به طور خلاصه می‌توان گفت که تجزیه مقادیر منفرد روشی است که یک ماتریس را به سه ماتریس دیگر تجزیه می‌کند. برای مثال، ماتریسA  را می‌توان به صورت زیر نوشت:

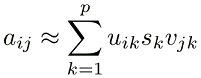
رابطه (۱)

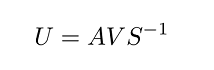
که در آن:

* A یک ماتریس m×n
* U یک ماتریس متعامد m×m
* S یک ماتریس قطری m×n
* و  V یک ماتریس متعامد  n×n است.

در رشته علوم کامپیوتر یکی از چالش‌های متداول در یادگیری ماشیـن، وجود چند صد متغیر اسـت، در حالی که بسیاری از الگوریتم‌های یادگیری ماشین، اگر با تعدادی بیش از یک مقدار مشخص متغیر کار کنند، با شکست مواجه می‌شوند. این موضوع، استفاده از تجزیه مقادیر تکین را برای کاهش متغیر در یادگیری ماشین ضروری می‌کند.

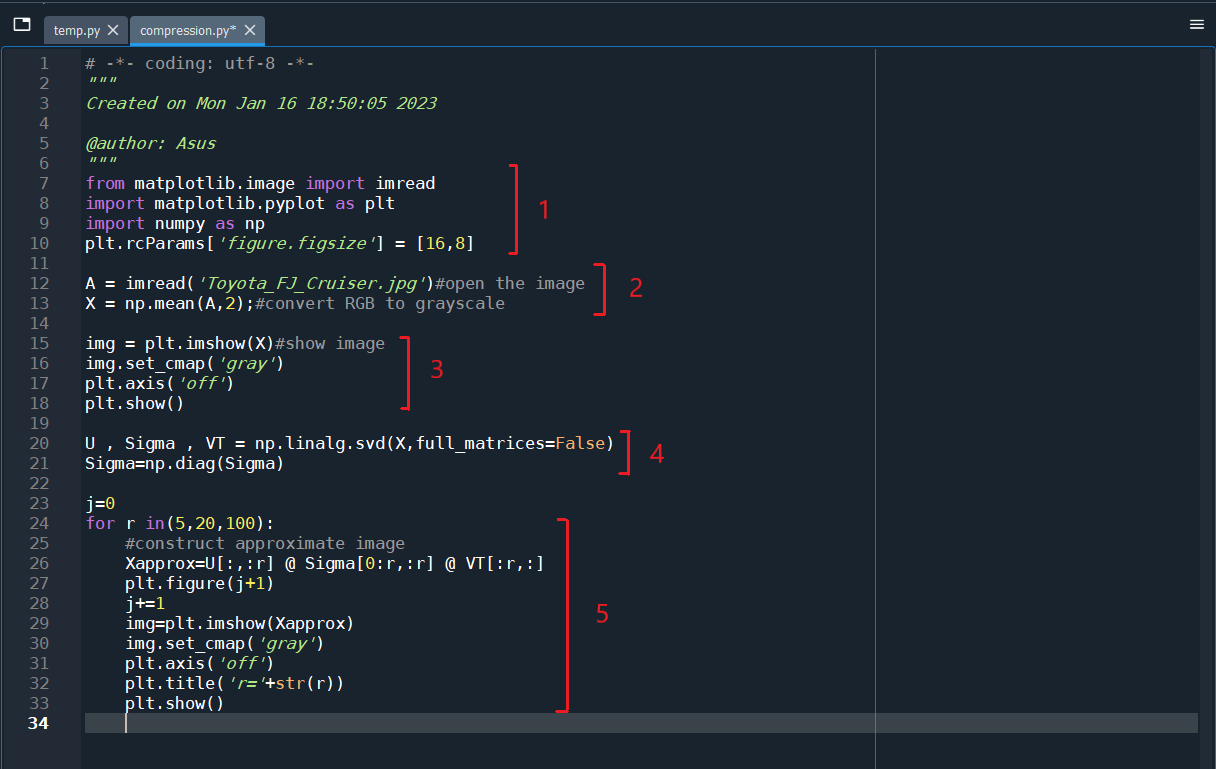
اگر تعداد و انواع مناسبی از ویژگی‌ها برای حل یک مساله خاص به الگوریتم‌های یادگیری ماشین داده شود، این الگوریتم‌ها به خوبی کار می‌کنند. اما، در صورتی که تعداد ویژگی‌ها (متغیرها) بسیار زیاد باشد، اغلب الگوریتم‌های یادگیری ماشین در حل مسئله دچار مشکل می‌شوند، زیرا با مسئله داده‌های ابعاد بالا (High Dimensional Data)مواجه خواهیم بود. در اینجا است که بحث کاهش ابعاد (Dimensionality Reduction) مطرح می‌شود.



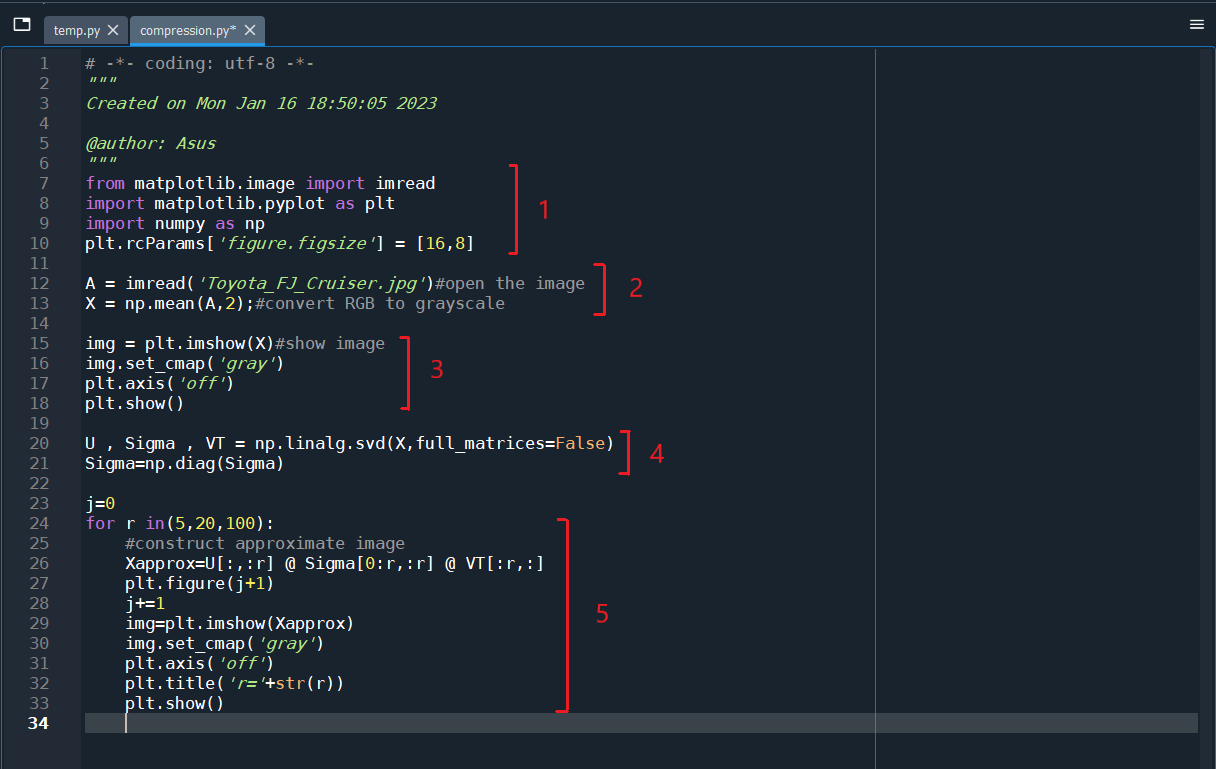
در معادله بالا یک SVD با کاهش تعداد مقادیر تکین می‌تواند یک ماتریس را بسیار دقیق تقریب بزند. از این ویژگی می‌توان برای فشرده‌سازی داده‌ها با فرم‌های کوتاه شده U، S ،V  به جای A  و برای کاهش متغیر از جایگزینیA  با U  استفاده کرد. البته در پایان، بر اساس معادله  باید با ضرب S  و V  نتایج را به دستگاه مختصات اصلی تبدیل کرد.

پیاده سازی فشرده سازی تصویر :

این عملیات در 5 مرحله با استفاده از کتابخانه های ریاضی پایتون بصورت زیر پیاده سازی شده است :



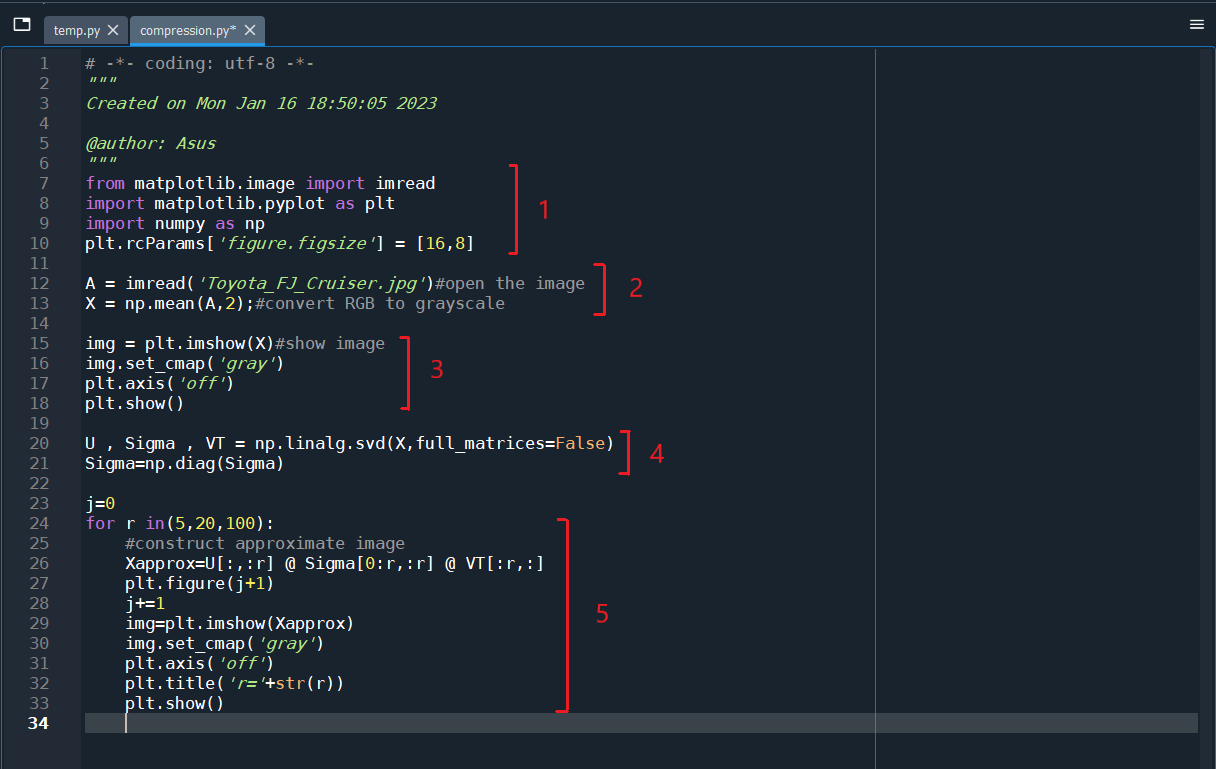
در مرحله اول با ضمیمه کردن کتابخانه های Matplotlib و Numpy شروع میکنیم که به ترتیب از Matplotlib برای نشان دادن و رسم کردن تصاویر فشرده شده و از Numpy برای استفاده از فرمول های جبرخطی مثل SVD و استخراج قطر اصلی (diag) استفاده خواهد شد .



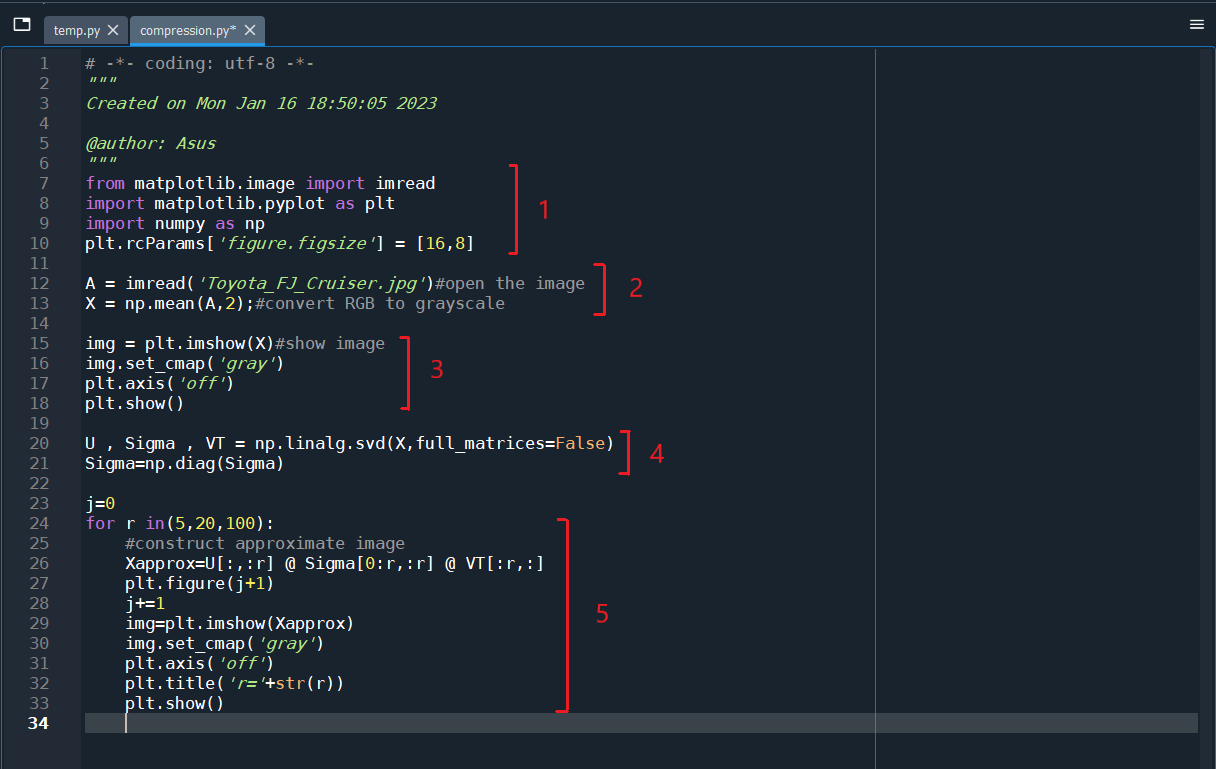
در اسکریپت بالا، از ویژگی rcParams در ماژول plt و ارسال مقدار figure.figsize به عنوان پارامتر استفاده کرده‌ایم که فهرستی حاوی عرض و ارتفاع پیش‌فرض نمودار را برمی‌گرداند. ایندکس اول عرض و ایندکس دوم ارتفاع است.

در مرحله دوم فایل تصویر را (که باید نام تصویر دلخواه را به عنوان ورودی به آن داد ) میخوانیم و در پارامتر A ذخیره میکنیم .

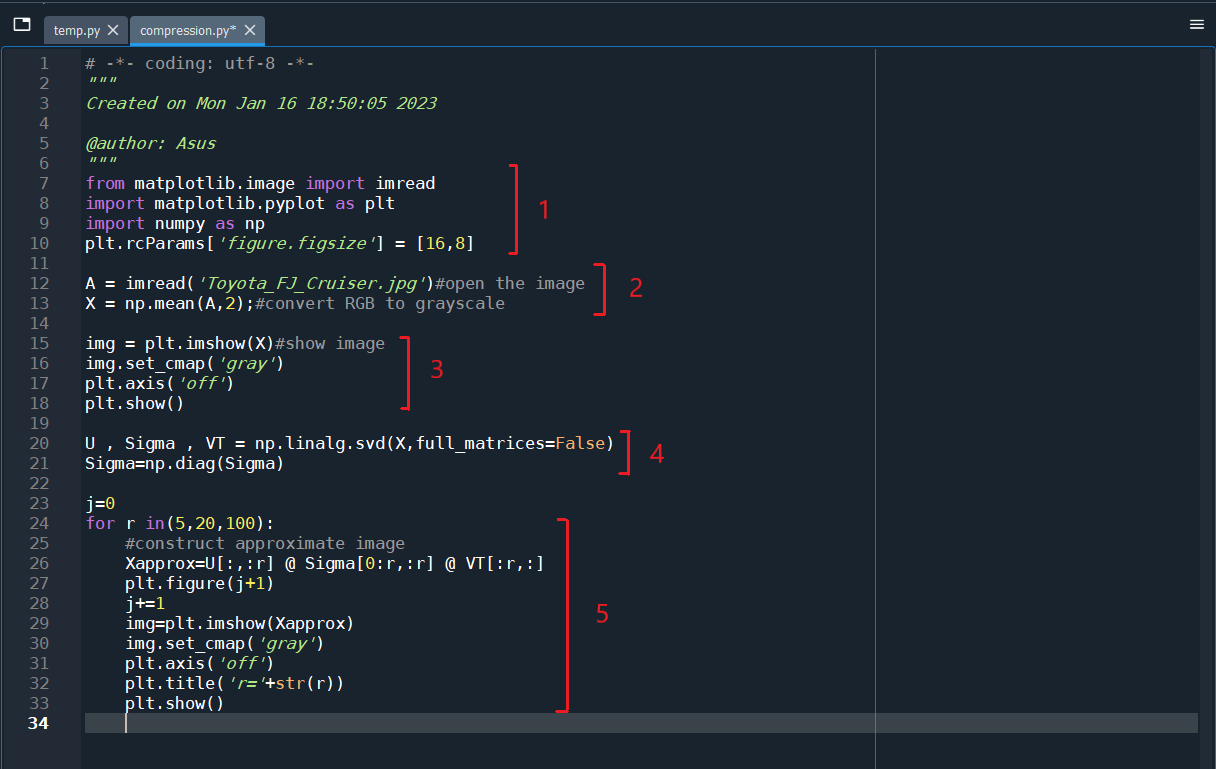
سپس با تابع Mean میتوانیم از سه عدد رنگ های قرمز , سبز و آبی RGB هر پیکسل که بین صفر و 255 می باشد میانگین بگیریم که بعدا رنگ تصویر را به حالت سفید سیاه دربیاوریم و محاسبات آسان تری داشته باشیم و آن را به شکل زیر در 2 بعد ذخیره می کنیم .



در مرحله سوم تصویر پردازش شده را به متغیر img می دهیم . میانگین اعدادی که از تصویر رنگی به دست اوردیم را به رنگ خاکستری متناظر میکنیم تا حالت سفید سیاه اجرا شود . سپس محور های افقی و عمودی تصویر را خاموش میکنیم که نمایش داده نشود و تصویر را نمایش می دهیم .



در مرحله چهارم وارد عملیات ریاضی شده و عکس ماتریس شده را به تابع SVD می دهیم تا با فرمول های LinAlg ( Linear Algebra) بتواند مقادیر U , S , VT را حساب کند و آن هارا جایگذاری کند .



برای در نظر گرفتن ماتریس های غیرمربعی حالت دیفالت تابع کهTrue میباشد را به حالت full\_matrices=False تغییر می دهیم . سپس قطر اصلی سیگما را در متغیر بصورت ارایه یک بعدی ذخیره میکنیم .

در مرحله آخر حلقه ای درست می کنیم که به ازای رتبه های دلخواه مان فشرده سازی را پیاده سازی کند که تفاوت تقریب با رتبه های مختلف را متوجه بشویم :

