

دانشگاه صنعتي امیرکبیر  
(پلی تکنیک تهران)

دانشكده مهندسی برق

ﮔﺰﺍﺭﺵکار تمرین عملی چهارم

مقدمه ای بر یادگیری ماشین

SVM

نگارش

علی بابالو

استاد راهنما

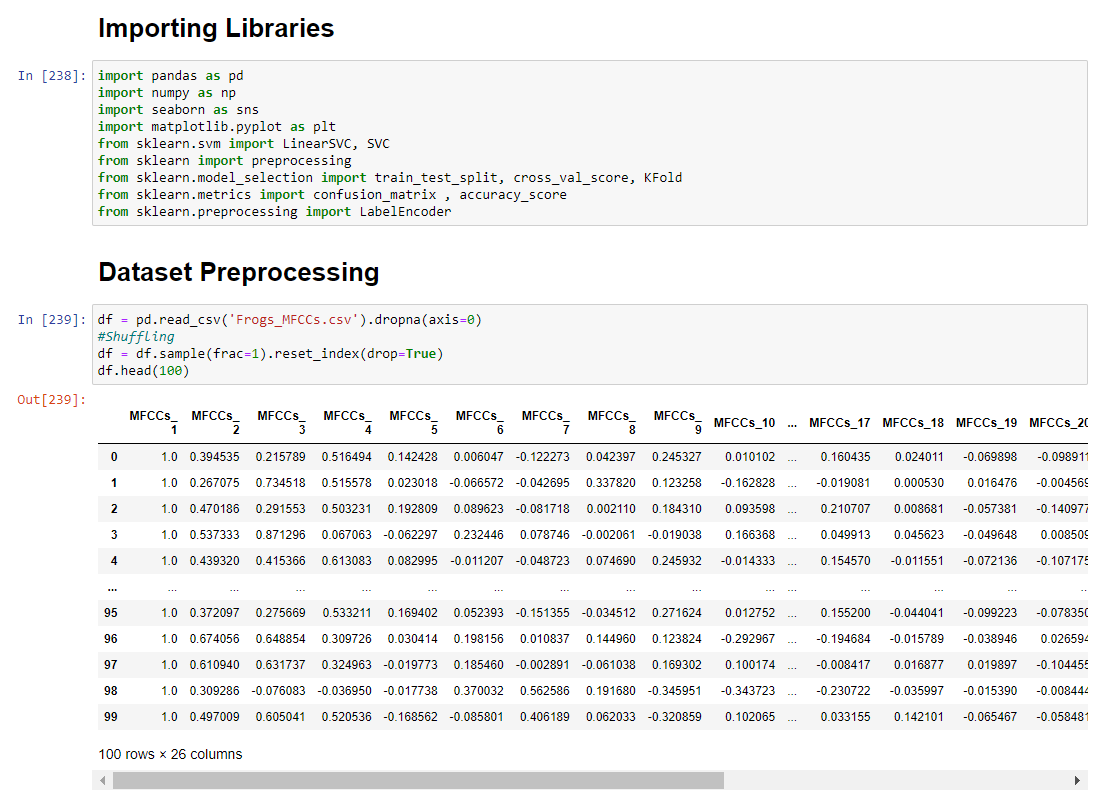
استاد ساناز سیدین

دی ماه 1140

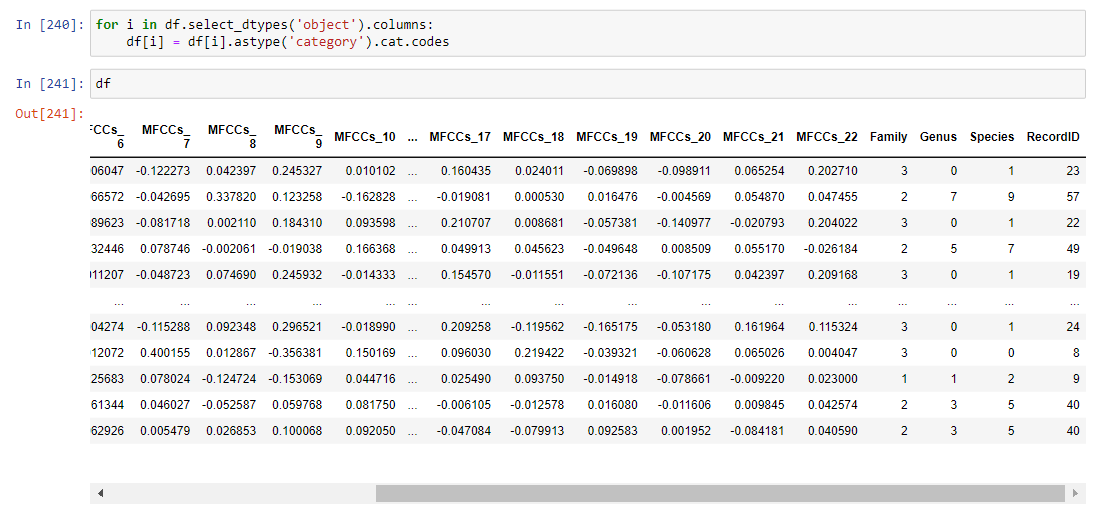
* **اضافه کردن کتابخانه**

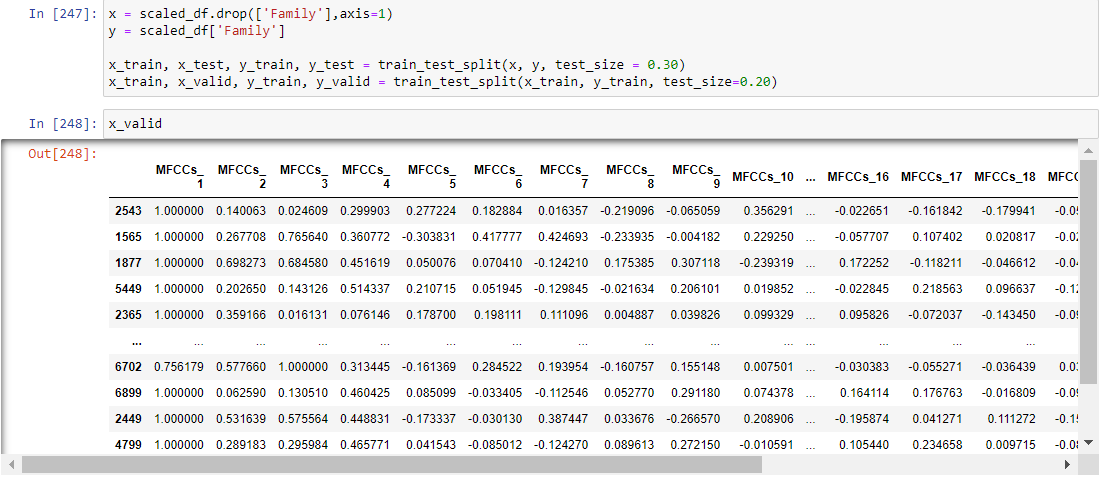
در ابتدا پس از کتابخانه های همیشگی پانداز و نامپای از کتابخانه sklearn و svm را ایمپورت می­کنیم. برای جدا کردن دیتاست به تست و ترین train\_test\_split را اضافه می­کنیم همچنین برای محاسبه کراس ولیدیشن accuracy\_score, cross\_val\_score را اضافه میکنیم. برای ماتریس به همریختگی هم confusion\_matrix را اضافه می­کنیم.

* **خواندن دیتاست**



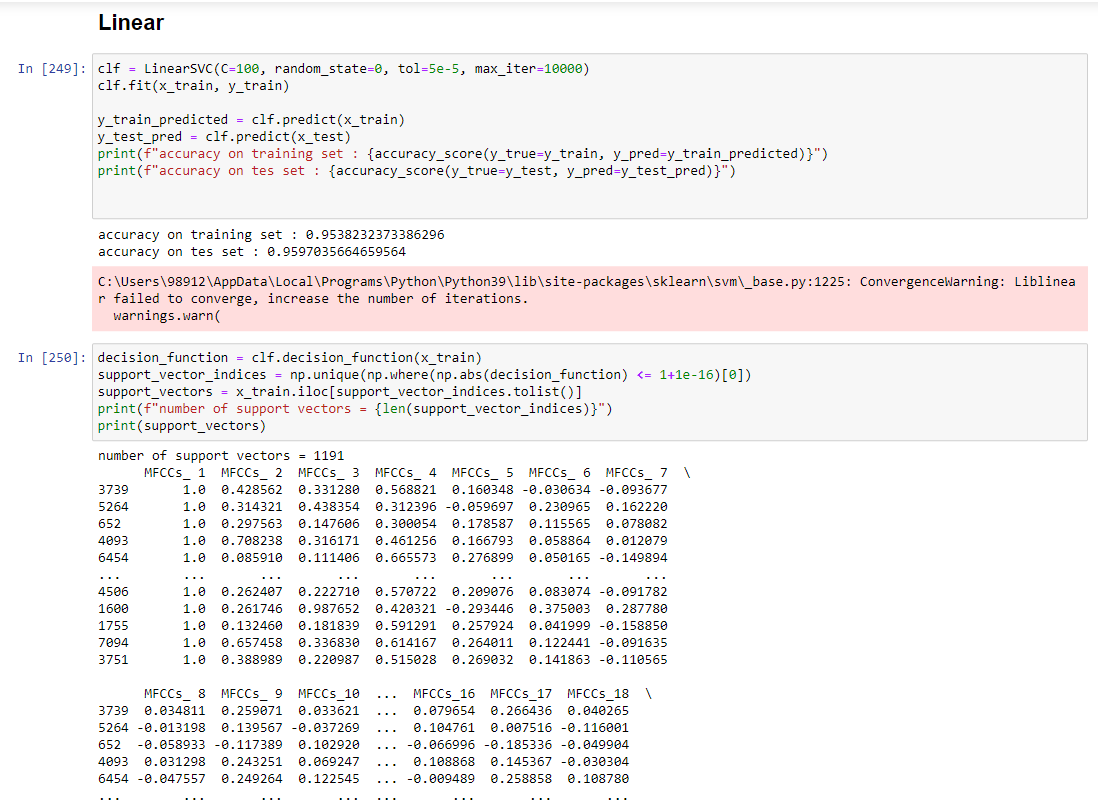
پس از خواندن دیتاست با پانداز ستون هایی از دیتاست که مقادیرشان به صورت آبجکن می­باشد را با دستور cat code به عدد تبدیل می­کنیم که از این کد در گزارش های قبلی استفاده کردیم و از توضیح اضافه می­پرهیزیم.

سپس دیتاست را با استفاده از تابع train\_test\_split به دیتا های آموزش و ولیدیشن و تست تقسیم می­کنیم.



* **Linear**

برای قسمت linear از تابع linearSVC استفاده می­کنیم و مدل را فیت می­کنیم و سپس با استفاده از decision\_function ساپورت وکتور هارا پیدا می­کنیم که نتایج در زیر قابل مشاهده هستند.



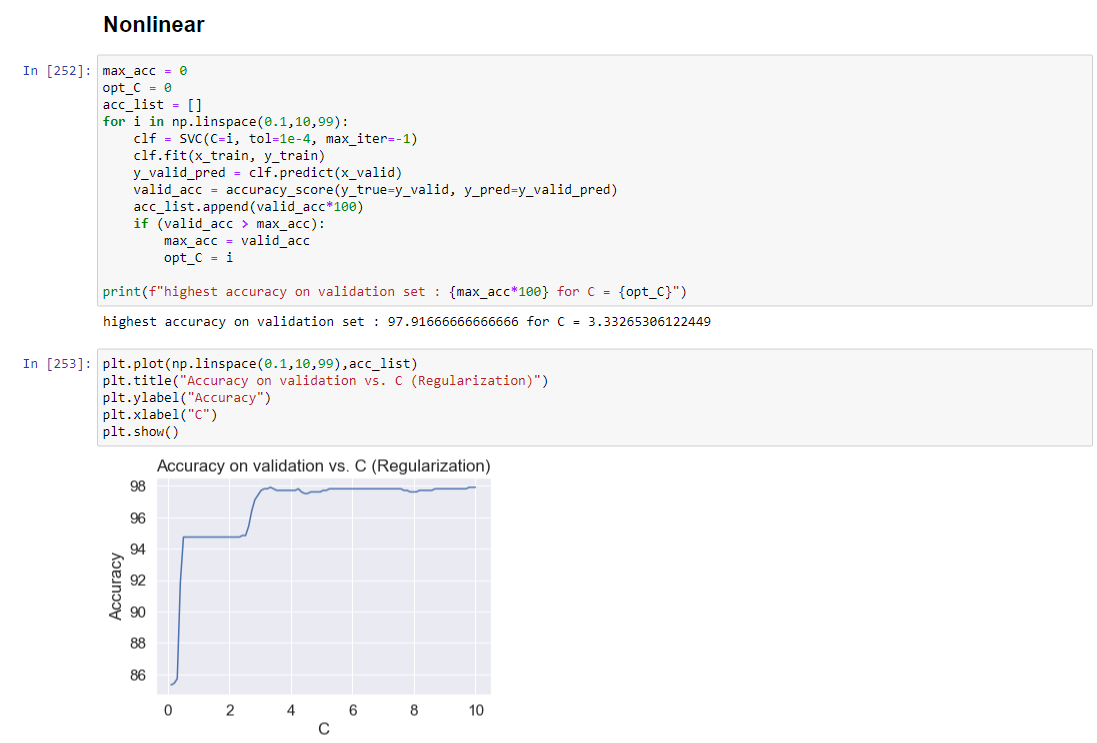
در پایین هم confusion matrix را مشاهده می­کنید.

Graphical user interface, application, Teams

Description automatically generated

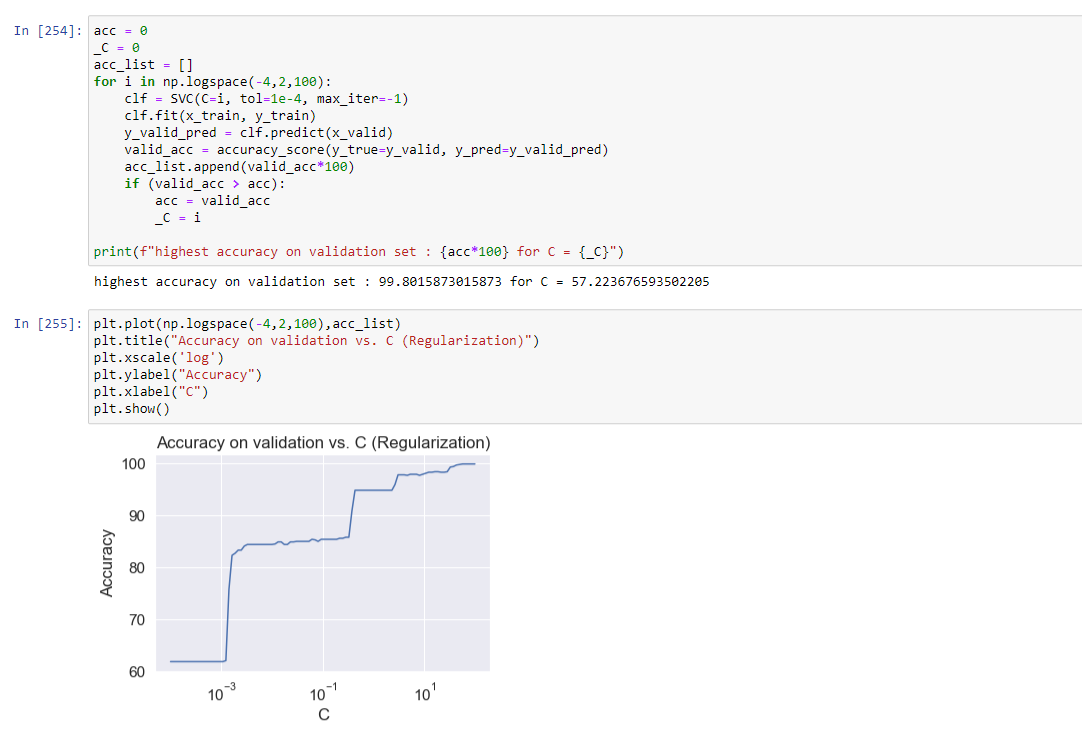
* **Soft margin**

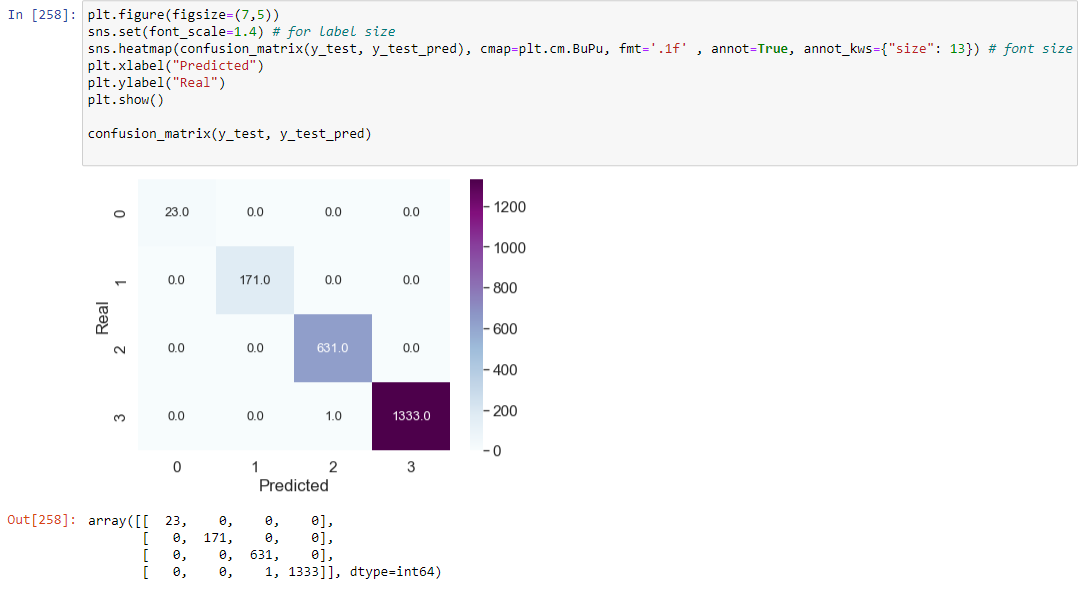
برای این قسمت خواسته شده تا بهترین مدل را پیدا کنیم برای همین ابتدا در یک محدوده معین مقدار اوپتیموم C را بدست می­اوریم تا از آن در فیت کردن مدلمان بهره ببریم. در مرحله اول دریک بازه خطی مقدار ماکسیمم دقت را بدست آوردیم. برای برست آوردن مقدار بهینه ابتدا در این بازه خطی حرکت می­کنیم و مدل را فیت می­کنیم و با دیتای ولیدیشن مدل را پردیکت می­کنیم سپس اگر دقت نقطه بعدی از قبلی بهتر بود ( دقت بیشتری داد) آن نقطه را به عنوان بهترین نقطه دقت و اوپتیمم C قرار می­دهیم.



اما برای اینکه محدوده بیشتری را پوشش دهیم از محدوده اگاریتمی استفاده کردیم برای اینکار از logspace از فاصله ۰.۰۰۰۴ تا ۱۰۰ را ۱۰۰ بازه کردیم و مقدار اوپتیمم C که بیشترین دقت را بدهد محاسبه کردیم. سپس با این مقدار C مدل را فیت سپس پردیکت کرده و دقت های لازم را محاسبه کردیم، برای پیدا کردن ساپورت وکتور ها از آبجکت support\_vector\_ استفاده کردیم که نتایج را مشاهده می­کنید.

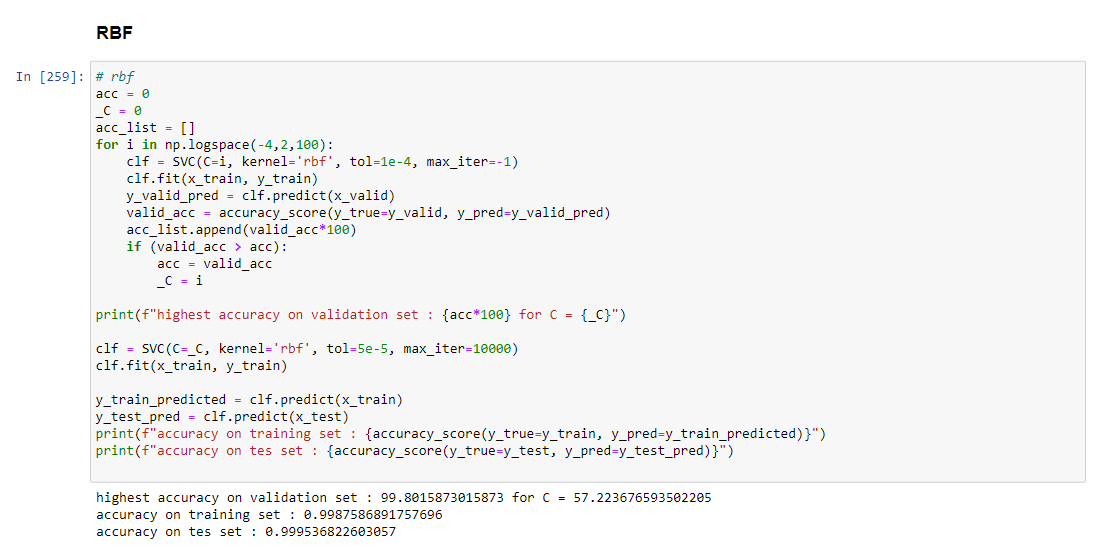
Graphical user interface, text, application, email

Description automatically generatedدر نهایت ماتریس به هم ریختگی را مشاهده می­کنید.



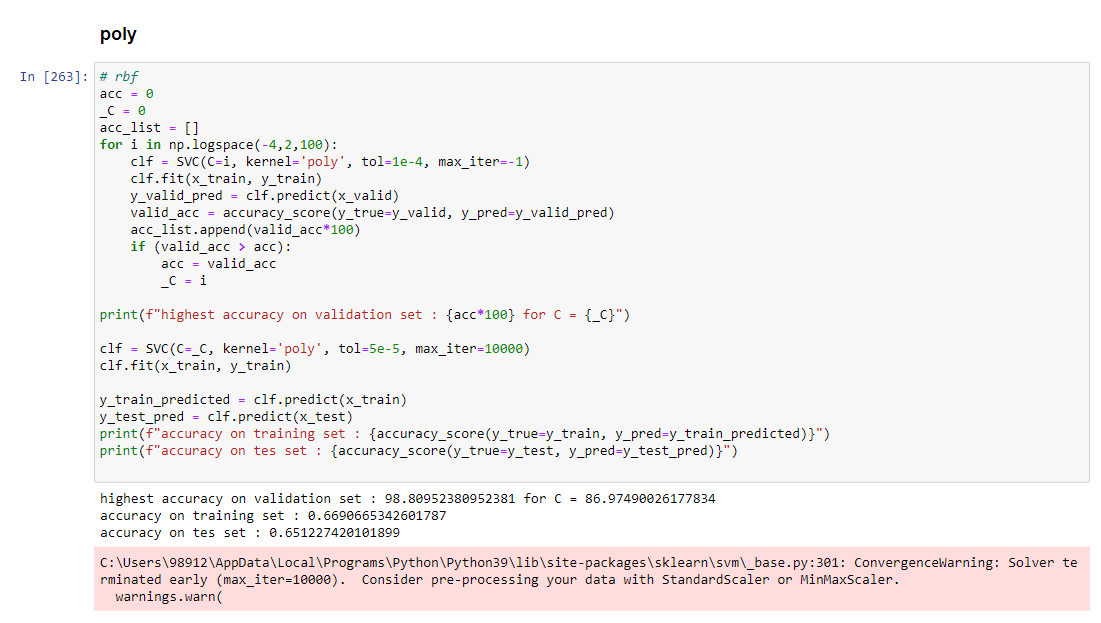
* RBF

در این مرحله همان کار های مرحله قبل را انجام می­دهیم اما کرنل را برابر rbf می­گذاریم:



* **Poly**

مانند قسمت قبل در این مرحله کرنل را poly قرار می­دهیم:

* **cross val**

برای این قسمت از همان تابع cross val که در تمرین بیزین استفاده کردیم بهره می­­بریم:

