# تمرین سری اول درس یادگیری ماشین

|  |
| --- |
| در محیط برنامه نویسی پایتون برنامه ای بنویسید که از دیتاست iris که حاوی اطلاعات سه نوع گیاه میباشد استفاده کند و 70 درصد داده ها را به عنوان داده آموزشی و 30 درصد داده ها به عنوان داده تست درنظربگیرد و با استفاده از دو الگوریتم LDA و QDA به طور جداگانه از داده های اموزشی برای آموزش شبکه و از داده های تست برای طبقه بندی استفاده کند.  به عنوان پیش پردازش از روش standardization استفاده کند.  در نهایت هم نتایج دو الگوریتم LDA و QDA را به طور جداگانه ارزیابی کرده و پارامترهای accuracy و loss را محاسبه کرده و چاپ کند.  یک فایل حاوی کدهای برنامه و گزارش تهیه کنید و از طریق ایمیل ارسال کنید. |

## تعریف کردن یک تابع برای محاسبه accuracy و loss

ابتدا یک تابع تعریف می‌کنیم تا به ازای هر لیبیلی که ماشین آن را درست پیش بینی می‌کند، یک عدد به متغیر Nt اضافه کند. سپس با تقسیم کردن Nt بر N، دقت یا Accuracy را حساب می‌کنیم و آن را Accurcy نام گذاری می‌کنیم. با کم کردن Accurcy از یک هم مقدار Loss را محاسبه می‌کنیم.

## آماده کردن فضای برنامه نویسی با فراخوانی کتابخانه‌های لازم

در گام بعدی کتابخانه‌های numpy و sklearn را با دستور import فراخوانی می‌کنیم.

## فراخوانی دیتاست

در گام بعدی، دیتا ست Iris را از بین دیتاست‌های کتابخانه sklearn فراخوانی و import می‌کنیم. سپس با استفاده از دستور X,L = load\_iris(return\_X\_y=True) لیبل‌ها (برچسب‌ها) و ویژگی‌های این دیتاست را فراخوانی می‌کنیم.

## تقسیم داده‌ها به دو بخش train و test

در مرحله بعد با استفاده از دستور train\_test\_split داده‌ها را به دو بخش تقسیم می‌کنیم. 70 درصد داده‌ها را برای آموزش یا train مدل و 30 درصد باقی مانده را برای آزمون یا test مدل برمی‌داریم.

## Standardization

سپس به کمک standardization داده‌هایمان را پیش پردازش می‌کنیم. یعنی داده‌هایمان را بین 0 و 1 می‌آوریم.

## لود کردن LDA Classifier

LDA Classifier یا مدل Linear Discriminant Analysis، مدلی است که قبلا در کتابخانه sklearn تعریف شده است. ما فقط آن را با دستور import فراخوانی می‌کنیم و مدلمان را با استفاده از این Classifier تعریف می‌کنیم و نامش را Model\_LDA می‌گذاریم.

## Train کردن مدل

به کمک دستور fit، مدلمان که نامش را Model\_LDA گذاشتیم train می‌کنیم یا آموزش می‌دهیم.

## Test کردن مدل

با استفاده از دستور predict مدلمان را روی داده‌های تست (30 درصد باقی مانده داده‌ها)، ارزیابی می‌کنیم.

## محاسبه accuracy و loss

به کمک تابعی که در اول کد تعریف کردیم، دقت و loss مدل را محاسبه می‌کنیم.

## پرینت نتایج

با استفاده از دستور print، Accuracy و loss مدل LDA را محاسبه می‌کنیم و نتایج آن را چاپ می‌کنیم.

## لود کردن QDA Classifier

این classifier هم قبلا در کتابخانه sklearn تعریف شده است. ما با استفاده از دستور import QuadraticDiscriminantAnalysis آن را فراخوانی می‌کنیم.

## Train کردن مدل دوم

به کمک دستور fit، مدلمان که نامش را Model\_QDA گذاشتیم، train می‌کنیم یا آموزش می‌دهیم.

## Test کردن مدل دوم

با استفاده از دستور predict مدل دوممان را روی داده‌های تست (30 درصد باقی مانده داده‌ها)، ارزیابی می‌کنیم.

## ارزیابی مدل دوم

به کمک تابعی که در اول کد تعریف کردیم، دقت و loss مدل QDA را هم محاسبه می‌کنیم.

## پرینت نتایج

با استفاده از دستور print، Accuracy و loss مدل QDA را محاسبه می‌کنیم و نتایج آن را چاپ می‌کنیم.

## نتیجه نهایی:

بار اول که کد را run کردم، دقت مدل LDA 0.9777 و مقدار Loss آن 0.0222 و دقت مدل QDA، 0.9555 و Loss آن 0.0444 محاسبه شده بود. بعد از run کردن مجدد کدها، دقت هر دو مدل 0.9777 محاسبه شد. در نهایت بعد از چند بار run کردن کدها، دقت هر دو مدل برابر 1.0 و loss آن‌ها برابر با 0.0 شد.

