

تمرین سری دوم درس یادگیری ماشین

در محیط برنامه نویسی پایتون برنامه‌ای بنویسید که از دیتاست `data.csv` که حاوی اطلاعات درباره بیماری سرطان سینه می‌باشد استفاده کند و داده‌ها را طبقه بندی کرده و نتیجه `accuracy` را در حالت‌های مختلف زیر گزارش کنید. به عنوان پیش پردازش از روش `standardization` استفاده کند.

الف) استفاده از الگوریتم `KNN` و در نظر گرفتن 70 درصد داده‌ها به عنوان داده آموزشی و 30 درصد داده‌ها به عنوان داده تست.

ب) استفاده از الگوریتم `KNN` و در نظر گرفتن روش `K-fold (K=10)`

ج) استفاده از الگوریتم `SVM` و در نظر گرفتن 70 درصد داده‌ها به عنوان داده آموزشی و 30 درصد داده‌ها به عنوان داده تست

د) استفاده از الگوریتم `SVM` و در نظر گرفتن روش `K-fold (K=10)`

1. ایمپورت کردن کتابخانه‌های مورد نیاز

در قدم اول کتابخانه‌های اصلی که برای این تمرین به آن‌ها نیاز داشتیم (`numpy`, `sklearn` و `pandas`) را `import` کردم.

2. لود کردن و خواندن دیتاست

در مرحله دوم، فایل اکسل دیتاست را فراخوانی کردم و آن را با دستور `pd.read_csv` لود کردم. سپس اطلاعات ستون لیبل‌ها را در `L` و اطلاعات سطرها و ستون‌های دیگر را در `X` ریختم.

3. قسمت الف تمرین و استفاده از مدل KNN

در این مرحله ابتدا دستور `train_test_split` را `import` کردم سپس با استفاده از آن داده‌ها را به صورت 70 درصد به عنوان داده آموزشی و 30 درصد داده تست تقسیم کردم. از دستوری `StandardScaler()` تحت عنوان `SC` برای پیش پردازش داده‌ها استفاده کردم.

سپس با دستور `from sklearn.neighbors import KNeighborsClassifier` مدل KNN را فراخوانی کردم و آن را تحت عنوان `Model` در کد استفاده کردم. با دستور `Model.fit` مدل را آموزش دادم سپس با دستور `Model.predict` آن را تست کردم.

در آخر با دستور `Model.score` مقدار `Accuracy` را حساب کردم. این مقدار برای قسمت الف تمرین `0.9707602339181286` بدست آمد.

4. قسمت ب تمرین استفاده از KNN و K-fold

ابتدا با دستور `KFold from sklearn.model_selection import` را فراخوانی کردم و آن را در ادامه `kf` نام گذاری کردم. مقدار `k` را 10 تعیین کردم. سپس یک لیست خالی تعریف کردم تا مقادیر `Accuracy` را در آن ذخیره کنم.

در قدم بعدی یک حلقه `for` نوشتم تا برای هر `fold` مراحل `train` و `test` را انجام بدهد و مقادیر `Accuracy` را در لیست بالا ذخیره کند. در آخر مقادیر `Loss` , `Accuracy` تمامی `fold` ها را پرینت کردم اما میانگین آن‌ها را هم حساب کردم. میانگین این مقادیر برابر با `0.9665726817042606` شد.

5. قسمت ج تمرین استفاده از الگوریتم SVM

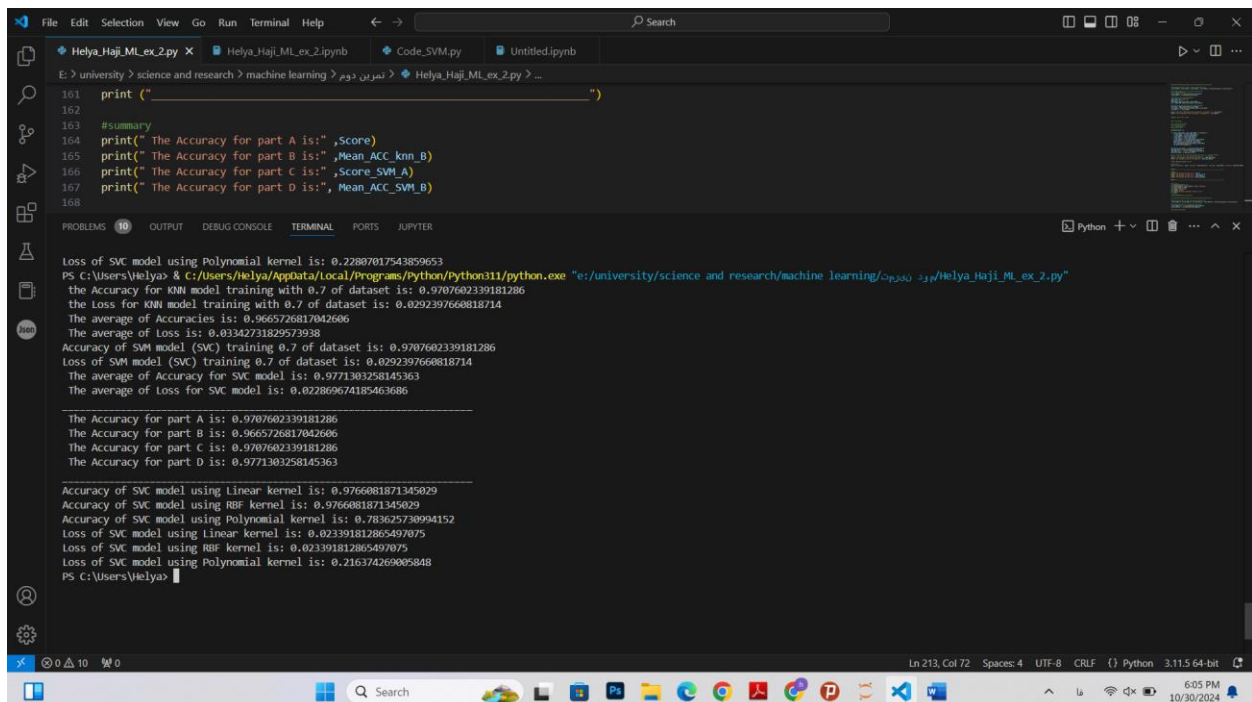
در این مرحله هم مانند قسمت الف، داده‌ها را به دو قسمت `train` و `test` تقسیم کردم. سپس با دستور `from sklearn.svm import SVC` مدل `SVC` (همان `SVM`) را فراخوانی کردم و مدل را با دستور `fit` آموزش دادم. در آخر با دستور `Model.score` `Accuracy` مدل را بررسی کردم که مقدار آن برابر با `0.9707602339181286` است.

6. قسمت د تمرین استفاده از SVM و K-fold

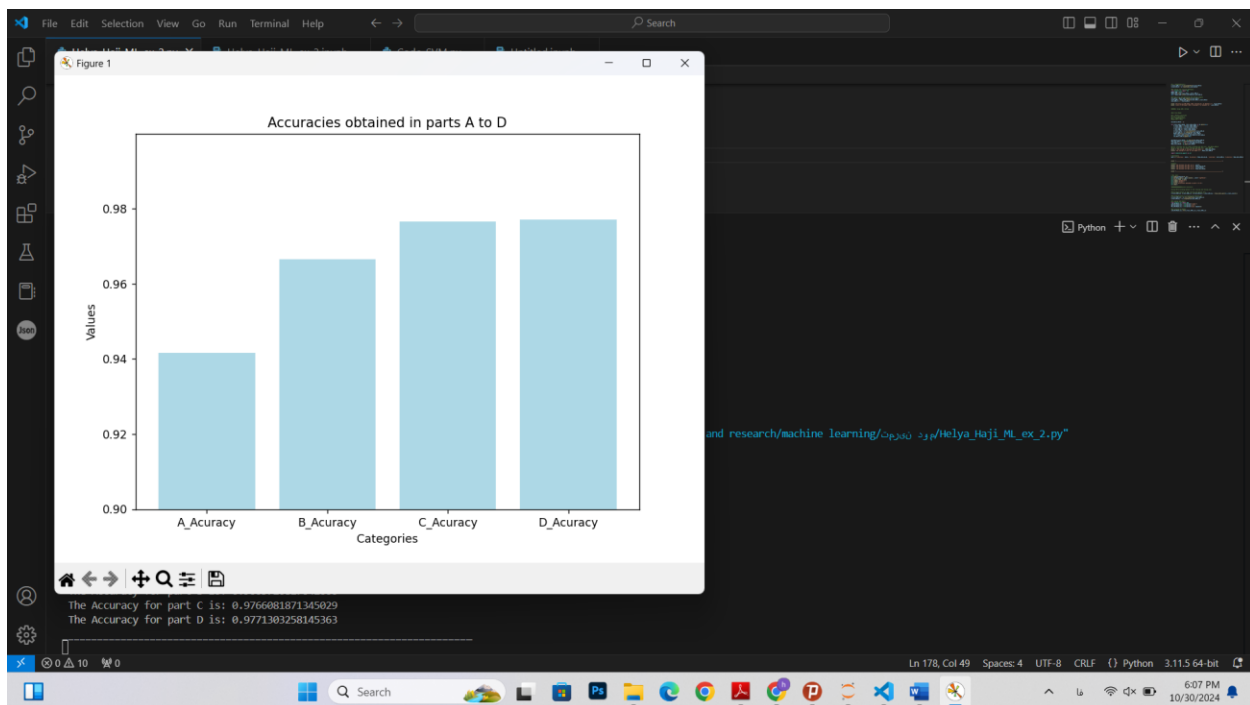
از k-fold استفاده کردم و مقدار k را برابر با 10 قرار دادم. سپس یک حلقه نوشتم تا Accuray را برای هر fold محاسبه کند. در آخر میانگین Accuray را با دستور np.mean حساب کردم که برابر با 0.9771303258145363 شد.

7. نتیجه گیری و خلاصه کردن نتایج

در آخر مقدار Accuray بدست آمده در چهار بخش الف تا د را حساب کردم و با یک نمودار نتایج را نشان دادم. به طور کلی با هر بار ران کردن کدها عدد Accuracy تغییر کرد اما به طور کلی بیشترین دقت برای قسمت ج و قسمت د مسئله بدست آمد. پس می توانیم این طور نتیجه گیری کنیم که دقت مدل SVM از مدل KNN بیشتر است.



```
File Edit Selection View Go Run Terminal Help
Helya_Haji_ML_ex_2.py x Helya_Haji_ML_ex_2.ipynb Code_SVM.py Untitled.ipynb
E: > university > science and research > machine learning > تمرین دوم > Helya_Haji_ML_ex_2.py > ...
161 print (" ")
162
163 #summary
164 print(" The Accuracy for part A is:",Score)
165 print(" The Accuracy for part B is:",Mean_ACC_knn_B)
166 print(" The Accuracy for part C is:",Score_SVM_A)
167 print(" The Accuracy for part D is:", Mean_ACC_SVM_B)
168
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER
Python + Python 3.11.5 64-bit
Loss of SVC model using Polynomial kernel is: 0.22807017543859653
PS C:\Users\Welya & C:\Users\Welya\AppData\Local\Programs\Python\Python311\python.exe "e:/university/science and research/machine learning/تمرین دوم/Helya_Haji_ML_ex_2.py"
the Accuracy for KNN model training with 0.7 of dataset is: 0.9707602339181286
the Loss for KNN model training with 0.7 of dataset is: 0.0292397660818714
The average of Accuracies is: 0.9665726817042606
The average of Loss is: 0.03342731829573938
Accuracy of SVM model (SVC) training 0.7 of dataset is: 0.9707602339181286
Loss of SVM model (SVC) training 0.7 of dataset is: 0.0292397660818714
The average of Accuracy for SVM model is: 0.9771303258145363
The average of Loss for SVC model is: 0.022869674185463680
-----
The Accuracy for part A is: 0.9707602339181286
The Accuracy for part B is: 0.9665726817042606
The Accuracy for part C is: 0.9707602339181286
The Accuracy for part D is: 0.9771303258145363
-----
Accuracy of SVC model using Linear kernel is: 0.9766081871345029
Accuracy of SVC model using RBF kernel is: 0.9766081871345029
Accuracy of SVC model using Polynomial kernel is: 0.7836257380994152
Loss of SVC model using Linear kernel is: 0.023391812865497075
Loss of SVC model using RBF kernel is: 0.023391812865497075
Loss of SVC model using Polynomial kernel is: 0.216374269005848
PS C:\Users\Welya
```



8. راه حل دوم برای قسمت ج

برای قسمت‌های ج و سوال راه حل دیگری هم به ذهنم رسید. در روش دوم من از هر سه کرنل SVM برای آموزش دادن مدل استفاده کردم. در این حالت دقت مدل SVM با کرنل Linear از سایر کرنل‌ها بالاتر بود.

