فایل ضمیمه : hw1.ipynb

فایل داده ها data.txt را به صورت یک دیتا فریم از pandas باز می‌کنم. طبق اطلاعات داده شده این مجموعه دارای ۴۴۸۰ ردیف داده است که هر کدام دارای ۵۳۵ مقدار هستند. طبق فایل labels.txt عنوان هر ستون مشخص شده. ستون آخر برچسب داده هاست که مقداری بین ۱ تا ۴ دارد و برابر است با شماره ی کلاس آن داده.

1-neutral, 2-emotional, 3-mental and 4-physical

ستون اول هم شمارگان شخصی است که مورد آزمایش قرارگرفته است که در مجموع ۴۰ نفر هستند. این ستون صرفا یک ایندکس است و در مراحل آموزش از آن استفاده نمی‌کنم. پس فضای داده هایی که قرار است کلاس بندی شوند ۵۳۲ بعد (ویژگی) دارد و مساله یک کلاس بندی ۴ کلاسه هست. اولین قدم جدا کردن داده های تست و آموزش است. ابتدا داده ها را درهم و سپس به نسبت ۳۳ به ۶۶ تقسیم می‌کنم. این کار با train_test_split کتابخانه ی sklearn انجام شده. تمام مراحل آموزش با داده های x_train و y_train انجام شده و تمام گزارشاتی که برای هر مدل رفته برای هر مدل، از روی مجموعه ی x_test با مدل ساخته شده دسته بندی صورت گرفته و نتایج این دسته بندی y_predict با مقادیر واقعی y_test مقایسه و confusion matix و accuracy و precision و recall و f-measure ساخته شده. برای اینکه درکی هم از عملکرد مدل روی داده های آموزش داشته باشم یک بار براورد روی آموزش هم انجام دادم. این گزارش ها با مجموعه ی metrics از کتابخانه ی sklearn تولید شده.

تذکر : داده هایی که با آن کار میکنم در ستون ۴۱۷ یک مقدار داشت که به صورت نوع داده ی اشتباهی ذخیره شده بود و کل این ستون با اینکه نوع داده ی عددی پیوسته دارد به صورت رشته ای ذخیره شده. قبل از ادامه ی کار این را اصلاح کردم.

Decission Tree

اولین مدل ساخته شده روی داده ها clf1 یک درخت تصمیم است با معیار information gain است که در جهت بهترین شاخه سازی برای کاهش انتروپی درخت را می‌سازد. این مدل با توابع sklearn ساخته شده. عمق ۱۷ دارد. روی داده های آموزش به طور کامل fit شده و هیچ خطای کلاس بندی ندارد. در هر برگ حداقل ۱ داده جای گرفته و روی هر گره برای شاخه سازی حداقل ۲ داده وجود داشته اند و معیارهای هرس درخت کاملا حداقلی تنظیم شده اند. داده های تست را با clf1 دسته بندی می‌کنم. نتایج زیر به دست آمده:

```
              precision    recall  f1-score   support

           1       0.96      0.94      0.95       372
           2       0.87      0.87      0.87       363
           3       0.86      0.85      0.85       376
           4       0.97      0.99      0.98       368

    accuracy                           0.91      1479
   macro avg       0.91      0.91      0.91      1479
weighted avg       0.91      0.91      0.91      1479

[[351   4  17   0]
 [  4 316  36   7]
 [ 11  43 319   3]
 [  1   0   1 366]]
```

از روی confusion matix بیشترین خطای مدل مربوط به طبقه بندی های کلاس ۲ و ۳ به جای یکدیگر است. قطر اصلی ماتریس طبقه بندی های صحیح است که بیشترین تعداد را روی هر سطر و ستون می‌سازد و در نتیجه معیارهای percision و recall و f-measure با اندازه های قابل قبولی برای هر کلاس به دست آمده. دقت accuracy مدل هم برابر ۹۱ که نتیجه ی مناسبی هست.

Random Forest

اولین مدل ساخته شده روی داده ها clf2 یک جنگل تصادفی است که در آن ۱۰۰ درخت با معیار information gain به عنوان دسته بند وجود دارند و هر کدام با نمونه گیری bootstrap روی داده ها و روی ویژگی ها در جهت بهترین شاخه سازی برای کاهش انتروپی ساخته شده اند. ساخت درخت با استفاده از sklearn انجام شده. با براورد برچسب داده های تست با clf2 نتایج زیر به دست می‌آید.

```
              precision    recall  f1-score   support

           1       1.00      0.99      0.99       372
           2       0.98      0.99      0.98       363
           3       0.98      0.98      0.98       376
           4       1.00      1.00      1.00       368

    accuracy                           0.99      1479
   macro avg       0.99      0.99      0.99      1479
weighted avg       0.99      0.99      0.99      1479

[[368   0   4   0]
 [  0 360   3   0]
 [  0   9 367   0]
 [  0   0   0 368]]
```

دقت به دست آمده در بهترین حالت هست. به خصوص از مقایسه با clf1 یک یک درخت بود، اجتماعی از ۱۰۰ درخت تصادفی ساخته شده اشتباهات به طور قابل ملاحظه ای کاهش یافت. مدل clf2 روی داده های آموزش کاملا fit شده و بدون خطاست و روی داده های تست هم تقریبا همینطور است و با دقت ۹۹ کلاس بندی انجام می‌شود.

XGBoost

مدل سوم یا clf3 یک مدل XGBoost است که با کتابخانه ی xgboost ساخته شده. برای این مدل از پارامترهای پیش فرض استفاده شده و تعداد درخت ها ۱۰۰ ، نرخ یادگیری ۰,۳۰۰۰۰۰۰۱۲ هست و حداکثر عمق هر درخت ۶ است. هربار هر درخت به روش تکراری از روی درخت قبلی ساخته می‌شود طوری که درخت قبلی را بهبود بدهد. نتایج به صورت زیر است:

```
              precision    recall  f1-score   support

           1       1.00      0.99      1.00       372
           2       0.99      0.99      0.99       363
           3       0.98      0.99      0.99       376
           4       1.00      1.00      1.00       368

    accuracy                           0.99      1479
   macro avg       0.99      0.99      0.99      1479
weighted avg       0.99      0.99      0.99      1479

[[370   0   2   0]
 [  0 358   4   1]
 [  0   3 373   0]
 [  0   0   0 368]]
```

همان طور که قابل انتظار هم بود clf3 نسبت به هر clf2 هم وضعیت بهتری دارد. این مدل روی ۱۴۷۹ داده ی تست تنها ۹ داده را اشتباه دسته بندی کرده. این مدل بهبود یافته ی همان جنگل تصادفی است به صورتی که ویژگی های مثبت آن را دارد از جمله اینکه از اجماع رای چند دسته بند استفاده می‌کند که به صورت تصادفی روی نمونه هایی از ویژگی ها و داده ها ساخته شده اند و این مزیت را دارد که تنها درخت هایی را شامل می‌شود که در نمونه های ساخته شده یکدیگر را کامل می کنند. این مدل روی هر دو مجوعه ی آموزش و تست کامل fit شده.

SVM

مدل clf4 یک ماشین بردار پشتیبان است. با مقدار ضریب منظم سازی ۱۰ و برای انتقال داده ها به فضای جدید از تابع radial basis به عنوان kernel استفاده می‌شود. تلاش بر این است که در فضای کرنل داده ها به صورت خطی جداشدنی باشند. نتایج مدل به صورت زیر هستند :

	precision	recall	f1-score	support
1	0.32	0.81	0.46	372
2	0.31	0.05	0.08	363
3	0.36	0.07	0.12	376
4	0.48	0.52	0.50	368
accuracy			0.36	1479
macro avg	0.37	0.36	0.29	1479
weighted avg	0.37	0.36	0.29	1479

```
[[301   8  11  52]
 [255  17  17  74]
 [250  14  27  85]
 [140  16  19 193]]
```

و روی داده های آموزش داریم :

	precision	recall	f1-score	support
1	0.32	0.82	0.46	748
2	0.34	0.05	0.09	757
3	0.42	0.09	0.14	744
4	0.50	0.57	0.53	752
accuracy			0.38	3001
macro avg	0.40	0.38	0.31	3001
weighted avg	0.40	0.38	0.31	3001

خطای مدل هم روی داده های تست و هم آموزش بسیار زیاد است و به لحاظ کارکرد قابل مقایسه با مدل ها ی قبلی نیست. داده های مساله فاصله ی زیادی با فرض اولیه SVM دارند و مشحص است که پیچیدگی یک دسته بند خطی برای جداسازی این داده ها مناسب نیست. دقت دسته بندی clf4 برابر با 36 درصد به دست آمده. اگر به confusion matix مراجعه شود، مدل تمایل دارد بیشتر داده ها را در دسته ی ۱ جای بدهد و همین باعث می‌شود مقادیر recall و f-measure روی کلاس های ۲ و ۳ خیلی کم باشد. این مدل برای پیدا کردن داده هایی با برچسب ۲ و ۳ اصلا قابل اعتماد نیست. همین ویژگی باعث کم بودن precision روی کلاس ۱ شده که دقت رای برای داده هایی با برچسب ۱ را پایین می‌برد.

5 Fold Cross Validation Results

بین ۴ مدل ساخته شده بهترین نتایج مربوط به مدل XGBoodt بود که به نام clf3 ساخته شد. برای گزارش نهایی عملکرد این مدل روی این دادگان از روش k fold cross validation با cv=5 استفاده می‌کنم. نتایج زیر میانگین عملکرد ۵ مدل ساخته شده روی داده های x_train , y_train است:

```
fit_time
11.308441638946533

score_time
0.030693578720092773

test_accuracy
0.9830027731558513

test_recall_macro
0.9829835115004638

test_precision_macro
0.9832222462418407

test_f1_macro
0.9830225731153739

test_recall_weighted
0.9830027731558513

test_precision_weighted
0.9832524804465852

test_f1_weighted
0.9830472746431346
```

همچین اگر روی همین ۵ مدل ساخته شده پیش بینی انجام شود حاصل confusion_matrix به صورت زیر خواهد بود.

```
[[732   3  13   0]
 [  0 743  14   0]
 [  4  15 724   1]
 [  0   0   1 751]]
```

به ازای مقدار accuracy بازگردانده شده از روش 5-fold cross validation مدلی که بهترین نتیجه را دارد به عنوان مدل نهایی استفاده می‌کنم و پارامترهای این مدل نهایی (مدل clf) :

```
{'objective': 'multi:softprob',
 'use_label_encoder': True,
 'base_score': 0.5,
 'booster': 'gbtree',
 'colsample_bylevel': 1,
 'colsample_bynode': 1,
 'colsample_bytree': 1,
 'enable_categorical': False,
 'gamma': 0,
 'gpu_id': -1,
 'importance_type': None,
 'interaction_constraints': '',
 'learning_rate': 0.300000012,
 'max_delta_step': 0,
 'max_depth': 6,
 'min_child_weight': 1,
 'missing': nan,
 'monotone_constraints': '()',
 'n_estimators': 100,
 'n_jobs': 8,
 'num_parallel_tree': 1,
 'predictor': 'auto',
 'random_state': 0,
 'reg_alpha': 0,
 'reg_lambda': 1,
 'scale_pos_weight': None,
 'subsample': 1,
 'tree_method': 'exact',
 'validate_parameters': 1,
 'verbosity': None}
```

مقایسه نتایج به دست آمده با نتایج مقاله

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6960825/

در این مقاله به جای ساخت مدل با تمام ویژگی ها، چند مدل ساخته که هرکدام با انتخاب تعدادی برابر Nmax از ویژگی ها کار می‌کنند و بعد از ساخت و تیت مدل ها، مدلی با ترکیب سیگنال های (ECG+TEB+EDA) را که با حداکثر م ویژگی از این دسته ساخته شده به عنوان بهترین مدل با احتمال خطای ۲۲,۲ درصد معرفی و به عنوان دسته بند انتخاب می‌کند. در جدول ۶ confusion matrix این مدل گزارش شده که با توجه به آن مقادیر precision و recall و accuracy برای این

مدل محاسبه می‌شود. نتایج این مدل در سطر آخر جدول با نوشته شده. تعداد ویژگی هایی که مدل های این تمرین را ساختم ۵۳۳ تا بود و بهترین نتیجه من ۹۸,۴ درصدر درست بود و در این مدل، معیار درستی accuracy برابر ۷۷ درصد با فقط ۴۰ تا از ویژگی ها به دست آمده.

Method	Accuracy	Precision	Recall	f-measure
Decision tree	0.91	0.91	0.91	0.91
Random Forest	0.99	0.99	0.99	0.99
XGBoost	0.99 *Cv :* 0.983	0.99 *Cv :* 0.983	0.99 *Cv:* 0.983	0.99 *Cv :* 0.983
SVM	0.36	0.37	0.36	0.29
	0.7705	0.77085	0.77052	