

ncomms2022——多模态dementia诊断

一、任务目标

通过图像数据（核磁共振MRI影像）和非影像数据（主要是临床数据）对认知状态进行分类。

NC——正常认知

MCI——轻度认知障碍

AD——阿兹海默痴呆

nADD——非阿兹海默痴呆

ncomms2022项目主要将任务分为两块：一是COG任务（三分类），分类NC、MRI、DE；二是ADD任务（二分类），是对COG任务中的DE进行细分，AD和nADD。

二、环境配置

Pytorch>=1.10

Numpy>=1.19

TQDM>=4.31

Nibabel>=3.2

matplotlib>=3.3

scikit-learn>=0.23

scipy>=1.5.4

SHAP>=0.37

XGBoosh>=1.3.3

catboost>=0.24

三、数据集下载

1、ADNI

非影像数据：需要慢慢筛选。（网站上的名称和实际下载的文件名不一致，需要慢慢筛选）

```
The following files need to be downloaded and saved in this folder
ADAS_ADNI023.csv      APOERES.csv    FHQ.csv         MRIMETA.csv
ADASSCORES.csv        BLCHANGE.csv   GDSCALE.csv     NEUROBAT.csv
ADNI_DXSUM_PDXCONV.csv CDR.csv        ITEM.csv        NPI.csv
ADNIMERGE.csv         EXCLUSIO.csv  MEDHIST.csv     NPIQ.csv
ADNI_MMSE.csv         FAMXHPAR.csv  MOCA.csv        PTDEMOG.csv
ADNI_NEUROPATH_04_12_18.csv FAMXHSIB.csv  MODHACH.csv
ADNI_NEUROPATH_MRI_04_12_18.csv FAQ.csv        MRI3META.csv
```

- ☒ Select ALL tabular data (csv format)
- ☒ Select ALL documents and zip files [450.0 GB]
- ☐ Assessments
 - ☐ ALL Diagnosis
 - ☐ Diagnosis and Symptom Checklist [ADNI1.G0.2]
 - ☐ Diagnostic Summary - Baseline Clinical Features [ADNI1.G0.2.3]
 - ☐ Diagnostic Summary [ADNI1.G0.2.3]
 - ☐ ALL Neuropsychological
 - ☐ ADAS Sub-Scores and Total Scores [ADNI1] Version: 2022-10-05
 - ☐ ADSP Phenotype Harmonization Consortium (PHC) - Composite Cognitive Scales [ADNI1.ADN1.G0.2.3] Version: 2022-10-05
 - ☐ ADSP Phenotype Harmonization Consortium (PHC) - Composite Cognitive Scales - Brief CFCF [ADNI1.ADN1.G0.2.3] Version: 2022-10-05
 - ☐ ADSP Phenotype Harmonization Consortium (PHC) - Composite Cognitive Scales [ADNI1.G0.2.3] Version: 2022-10-05
 - ☐ Alzheimer's Disease Assessment Scale (ADAS) [ADNI1]
 - ☐ Alzheimer's Disease Assessment Scale (ADAS) [ADNI0.G0.2.3]
 - ☐ clinical Dementia Rating Scale (CDR) [ADNI1.G0.2.3]
 - ☐ Cognitive Change Index [ADNI0.2]
 - ☐ Cognitive Battery Documentation [ADNI0.2]
 - ☐ Cognitive Battery Results [ADNI0.2.3]
 - ☐ Cognitive Brief Battery [ADNI0.2]
 - ☐ Embic Correlation - Quantified Cognitive Processes Methods (PDF) [ADNI0.2.3] Version: 2023-11-28
 - ☐ Embic Correlation - Quantified Cognitive Processes [ADNI1.G0.2.3]
 - ☐ Embic Correlation - Digital Cognitive Biomarkers Methods (PDF) [ADNI0.2.3] Version: 2022-08-09
 - ☐ Embic Correlation - Digital Cognitive Biomarkers [ADNI1.G0.2.3] Version: 2022-08-09
 - ☐ Financial Capacity - Participant Self Report [ADNI0.2.3]
 - ☐ Everyday Cognition - Study Partner Report [ADNI0.2.3]
 - ☐ Everyday Capacity Inventory Short Form (ECISF) [ADNI0.3]
 - ☐ Functional Activities Questionnaire (FAQ) [ADNI1.G0.2.3]
 - ☐ Geriatric Depression Scale (GDS) [ADNI1.G0.2.3]
 - ☐ Item Level Data (ADAS-Cog, CDR, CDRSB, MMSE, etc.) [ADNI1.G0.2.3]
 - ☐ Mini-Mental State Examination (MMSE) [ADNI1.G0.2.3]
 - ☐ Modified Hachinski Ischemia Scale [ADNI1.G0.2.3]
 - ☐ Montreal Cognitive Assessment (MoCA) [ADNI1.G0.2.3]
 - ☐ Neurocognitive Inventory (NFI) [ADNI0.2.3]
 - ☐ Neuropsychiatric Inventory (NPI) [ADNI1.G0.2.3]

The screenshot shows the IDA Pro interface with the 'Data Collections' window open. The window title is 'Collection: ADNI1:Complete 3Yr 1.0T'. Below the title bar, there is a table of data collections. The table has columns: 'Collection', 'Name', 'Size', 'Type', 'Status', 'Description', 'Date', 'Version', 'Location', 'Accession', and 'Action'. The 'Collection' column is highlighted in blue. The table lists various data points, including 'ADNI1:Complete 3Yr 1.0T', 'ADNI1:Complete 3Yr 1.0T (S1)', 'ADNI1:Complete 3Yr 1.0T (S2)', 'ADNI1:Complete 3Yr 1.0T (S3)', 'ADNI1:Complete 3Yr 1.0T (S4)', 'ADNI1:Complete 3Yr 1.0T (S5)', 'ADNI1:Complete 3Yr 1.0T (S6)', 'ADNI1:Complete 3Yr 1.0T (S7)', 'ADNI1:Complete 3Yr 1.0T (S8)', 'ADNI1:Complete 3Yr 1.0T (S9)', 'ADNI1:Complete 3Yr 1.0T (S10)', 'ADNI1:Complete 3Yr 1.0T (S11)', 'ADNI1:Complete 3Yr 1.0T (S12)', 'ADNI1:Complete 3Yr 1.0T (S13)', 'ADNI1:Complete 3Yr 1.0T (S14)', 'ADNI1:Complete 3Yr 1.0T (S15)', 'ADNI1:Complete 3Yr 1.0T (S16)', 'ADNI1:Complete 3Yr 1.0T (S17)', 'ADNI1:Complete 3Yr 1.0T (S18)', 'ADNI1:Complete 3Yr 1.0T (S19)', 'ADNI1:Complete 3Yr 1.0T (S20)', 'ADNI1:Complete 3Yr 1.0T (S21)', 'ADNI1:Complete 3Yr 1.0T (S22)', 'ADNI1:Complete 3Yr 1.0T (S23)', 'ADNI1:Complete 3Yr 1.0T (S24)', 'ADNI1:Complete 3Yr 1.0T (S25)', 'ADNI1:Complete 3Yr 1.0T (S26)', 'ADNI1:Complete 3Yr 1.0T (S27)', 'ADNI1:Complete 3Yr 1.0T (S28)', 'ADNI1:Complete 3Yr 1.0T (S29)', 'ADNI1:Complete 3Yr 1.0T (S30)', 'ADNI1:Complete 3Yr 1.0T (S31)', 'ADNI1:Complete 3Yr 1.0T (S32)', 'ADNI1:Complete 3Yr 1.0T (S33)', 'ADNI1:Complete 3Yr 1.0T (S34)', 'ADNI1:Complete 3Yr 1.0T (S35)', 'ADNI1:Complete 3Yr 1.0T (S36)', 'ADNI1:Complete 3Yr 1.0T (S37)', 'ADNI1:Complete 3Yr 1.0T (S38)', 'ADNI1:Complete 3Yr 1.0T (S39)', 'ADNI1:Complete 3Yr 1.0T (S40)', 'ADNI1:Complete 3Yr 1.0T (S41)', 'ADNI1:Complete 3Yr 1.0T (S42)', 'ADNI1:Complete 3Yr 1.0T (S43)', 'ADNI1:Complete 3Yr 1.0T (S44)', 'ADNI1:Complete 3Yr 1.0T (S45)', 'ADNI1:Complete 3Yr 1.0T (S46)', 'ADNI1:Complete 3Yr 1.0T (S47)', 'ADNI1:Complete 3Yr 1.0T (S48)', 'ADNI1:Complete 3Yr 1.0T (S49)', 'ADNI1:Complete 3Yr 1.0T (S50)', 'ADNI1:Complete 3Yr 1.0T (S51)', 'ADNI1:Complete 3Yr 1.0T (S52)', 'ADNI1:Complete 3Yr 1.0T (S53)', 'ADNI1:Complete 3Yr 1.0T (S54)', 'ADNI1:Complete 3Yr 1.0T (S55)', 'ADNI1:Complete 3Yr 1.0T (S56)', 'ADNI1:Complete 3Yr 1.0T (S57)', 'ADNI1:Complete 3Yr 1.0T (S58)', 'ADNI1:Complete 3Yr 1.0T (S59)', 'ADNI1:Complete 3Yr 1.0T (S60)', 'ADNI1:Complete 3Yr 1.0T (S61)', 'ADNI1:Complete 3Yr 1.0T (S62)', 'ADNI1:Complete 3Yr 1.0T (S63)', 'ADNI1:Complete 3Yr 1.0T (S64)', 'ADNI1:Complete 3Yr 1.0T (S65)', 'ADNI1:Complete 3Yr 1.0T (S66)', 'ADNI1:Complete 3Yr 1.0T (S67)', 'ADNI1:Complete 3Yr 1.0T (S68)', 'ADNI1:Complete 3Yr 1.0T (S69)', 'ADNI1:Complete 3Yr 1.0T (S70)', 'ADNI1:Complete 3Yr 1.0T (S71)', 'ADNI1:Complete 3Yr 1.0T (S72)', 'ADNI1:Complete 3Yr 1.0T (S73)', 'ADNI1:Complete 3Yr 1.0T (S74)', 'ADNI1:Complete 3Yr 1.0T (S75)', 'ADNI1:Complete 3Yr 1.0T (S76)', 'ADNI1:Complete 3Yr 1.0T (S77)', 'ADNI1:Complete 3Yr 1.0T (S78)', 'ADNI1:Complete 3Yr 1.0T (S79)', 'ADNI1:Complete 3Yr 1.0T (S80)', 'ADNI1:Complete 3Yr 1.0T (S81)', 'ADNI1:Complete 3Yr 1.0T (S82)', 'ADNI1:Complete 3Yr 1.0T (S83)', 'ADNI1:Complete 3Yr 1.0T (S84)', 'ADNI1:Complete 3Yr 1.0T (S85)', 'ADNI1:Complete 3Yr 1.0T (S86)', 'ADNI1:Complete 3Yr 1.0T (S87)', 'ADNI1:Complete 3Yr 1.0T (S88)', 'ADNI1:Complete 3Yr 1.0T (S89)', 'ADNI1:Complete 3Yr 1.0T (S90)', 'ADNI1:Complete 3Yr 1.0T (S91)', 'ADNI1:Complete 3Yr 1.0T (S92)', 'ADNI1:Complete 3Yr 1.0T (S93)', 'ADNI1:Complete 3Yr 1.0T (S94)', 'ADNI1:Complete 3Yr 1.0T (S95)', 'ADNI1:Complete 3Yr 1.0T (S96)', 'ADNI1:Complete 3Yr 1.0T (S97)', 'ADNI1:Complete 3Yr 1.0T (S98)', 'ADNI1:Complete 3Yr 1.0T (S99)', 'ADNI1:Complete 3Yr 1.0T (S100)'. The 'Collection' column is highlighted in blue.

10. Δp (Pa) = 3.51 (28) - 4.12 (28) | about 1 hour = 1 hour

General: This document will display your account information and you can manage your account.

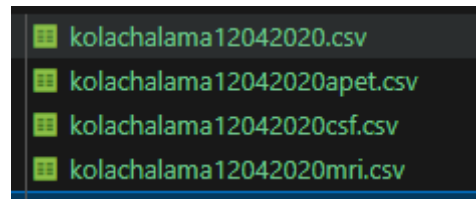
Zip Files

Metadata URL List

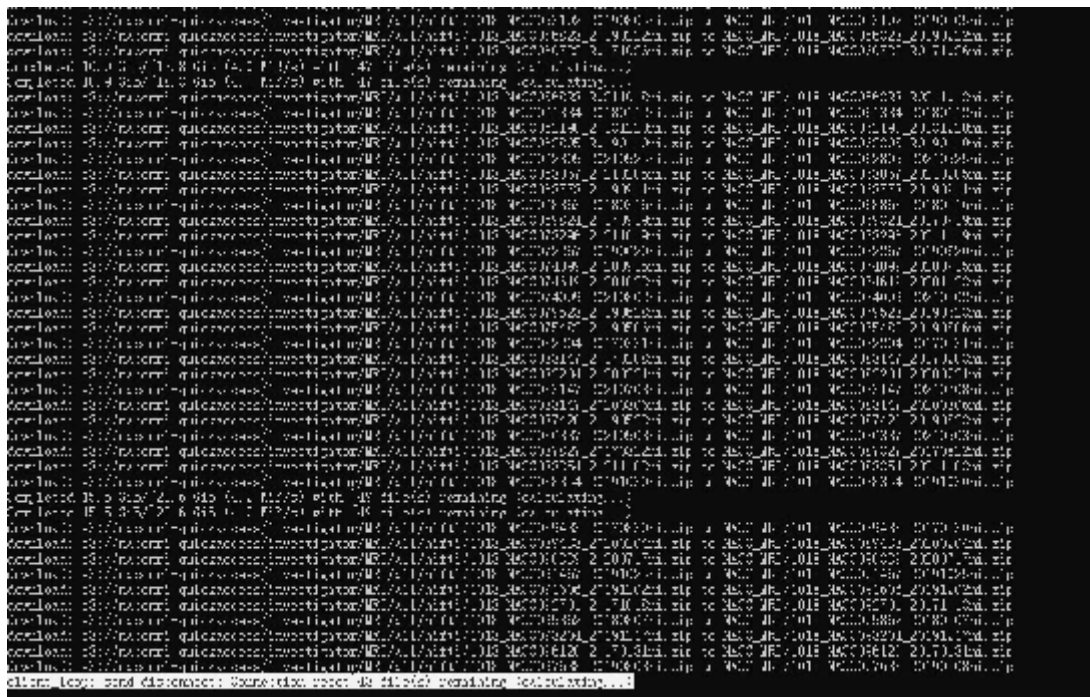
To avoid browser connection problems, learn [About Download Managers](#)

2、NACC

非影像数据：需要向机构申请



影像数据：数据量大（预计25G以上）且不知道是否与非图像数据匹配。原论文及项目中未提及具体如何选取。（目前下载失败）



四、数据预处理及训练集划分

1、进入derived_tables文件夹，根据不同数据集生成元数据。

```
run the 3 python scripts to generate derived meta information  
for MOXA, MMSE and diagnosis.
```

2、进入datasets_tables文件夹，为不同数据集准备最终的元表。

```
run the ADNI1.py to get the meta table for the ADNI1 data
```

```
> ADNI1  
> ADNI2  
> ADNI3  
> ADNIGO
```

```

class TableData:
    def __init__(self):
        self.datasetName = 'ADNI1'
        self.imageDir = 'data/ADNI1/'
        self.imageFileNameList, self.RIDList = self.get_filenames_and_IDs(self.imageDir)
        print(self.imageFileNameList)
        self.columnNames = []
        self.content = defaultdict(dict) # dictionary of dictionary; {RID: {colname1: val1, colname2: val2, ...}}

    def get_filenames_and_IDs(self, path):
        fullPathList = glob(path + '*.nii')
        print(fullPathList)
        fileNameList = [fullpath.split('/')[-1] for fullpath in fullPathList]
        IDList = [filename[3:15] for filename in fileNameList]
        RIDList = [id[-4:].lstrip("0") for id in IDList]
        return fileNameList, RIDList

```

其中需要以图片路径作为索引，选取记录元素，并得到列名

3、进入CrossValid文件夹，划分训练集、测试集、验证集。

```

this CrossValid folder will contain 5 fold cross validation split information

step1, run the combine.py to combine all subjects from all cohorts and output results in all.csv

step2, run the split.py to split the NACC into train, valid, test, and consider all other cohorts
as external testing, the data split results will be stored in folder cross0/ cross1/ etc

step3, run appendNonImage.py to fill up non-imaging information by table joining

```

combine.py将所有数据集整合到一个all.csv文件中。

split.py将all.csv划分为训练集、测试集、验证集。

appendNonImage.py将所有非图像元素加入到训练集、测试集、验证集中。（列名）

```

import csv

# tables = ['NACC', 'ADNI1', 'ADNI2', 'ADNI3', 'ADNIGO', 'NIFD', 'PPMI', 'AIBL', 'OASIS', 'FHS', 'Stanford']
tables = ['ADNI1', 'ADNI2', 'ADNI3']
column_names = ['path', 'filename', 'NC', 'MCI', 'DE', 'COG', 'AD', 'PD', 'FTD', 'VD', 'DLB', 'PDD', 'ADD', 'ALL', 'OTHER']

def read_csv_dict(content, csv_table):
    with open(csv_table, 'r') as f:
        reader = csv.DictReader(f)
        for row in reader:
            content.append(row)

content = []

for table in tables:
    if table == 'NACC':
        csv_path = '../dataset_table/NACC_ALL/' + table + '.csv'
    else:
        csv_path = '../dataset_table/' + table + '/' + table + '.csv'
    read_csv_dict(content, csv_path)

with open('all.csv', 'w', newline='') as csvfile:
    spamwriter = csv.writer(csvfile)
    spamwriter.writerow(column_names)
    for row in content:
        if row['NC'] == '1':
            row['ALL'] = '0'
        elif row['MCI'] == '1':
            row['ALL'] = '1'
        elif row['AD'] == '1':
            row['ALL'] = '2'
        elif row['ADD'] == '0':
            row['ALL'] = '3'
        spamwriter.writerow([row[col_name] if col_name in row else '' for col_name in column_names])

```

五、训练模型(NoImg)

准备好训练集、测试集、验证集后（目前仅有ADNI1中的1/10），训练CNN模型（由于没有NACC，只能COG任务）（test.ipynb）

```

from nonimg_model_wrappers import NonImg_Model_Wrapper, Fusion_Model_Wrapper
from utils import read_json

model = NonImg_Model_Wrapper(
    tasks=['COG'], # a list of tasks to predict
    main_config={"save_dir": "lookups/CrossValid/cross0/", "model_name": "CNN_baseline_new_cross0"}, # general configuration for the experiment
    task_config=read_json('nonimg_task_config.json'), # task specific configurations
    seed=1000)

model.train()
thres = model.get_optimal_thres() # get optimal threshold using validation dataset
model.gen_score(['test'], thres) # apply optimal threshold on test dataset and cache predictions

```

nonimg_task_config.json选取合适特征，保证特征维度匹配

```

CNN_baseline_new_cross0
  COG_shop_background4_shop_background_COG.png
  events.out.tfevents.1701591670.user-System-Product-Name.3959210.0
  events.out.tfevents.1701591724.user-System-Product-Name.3959210.1
  events.out.tfevents.1701768757.user-System-Product-Name.3979700.0
  events.out.tfevents.1701760922.user-System-Product-Name.3979915.0
  events.out.tfevents.1701769390.user-System-Product-Name.3980029.0
  events.out.tfevents.1701760440.user-System-Product-Name.3980138.0
  events.out.tfevents.1701769401.user-System-Product-Name.3980255.0
  events.out.tfevents.1701769544.user-System-Product-Name.3980359.0
  events.out.tfevents.1701760568.user-System-Product-Name.3980463.0
  events.out.tfevents.1701769715.user-System-Product-Name.3980602.0
  events.out.tfevents.1701769822.user-System-Product-Name.3980775.0
  events.out.tfevents.1701760848.user-System-Product-Name.3980879.0
  events.out.tfevents.1701769073.user-System-Product-Name.3980679.1
  events.out.tfevents.1701769883.user-System-Product-Name.3981011.0
  events.out.tfevents.1701760904.user-System-Product-Name.3981126.0
  events.out.tfevents.1701770000.user-System-Product-Name.3981243.0
  events.out.tfevents.1701770131.user-System-Product-Name.3981317.0
  events.out.tfevents.1701770151.user-System-Product-Name.3981452.0
  events.out.tfevents.1701770170.user-System-Product-Name.3981556.0
  events.out.tfevents.1701770210.user-System-Product-Name.3981678.0
  shop_COG_shop_background4_COG.png
  test_eval.csv
  valid_eval.csv

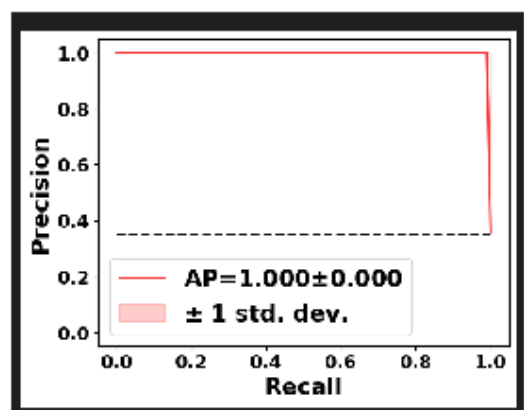
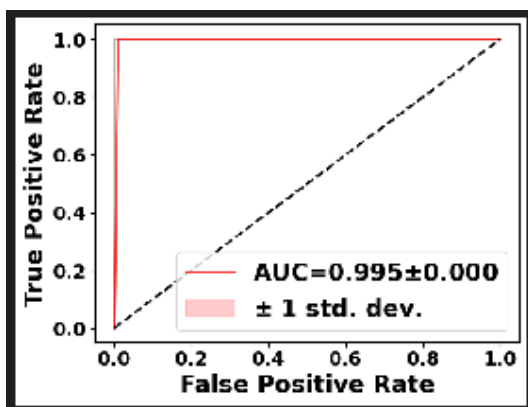
```

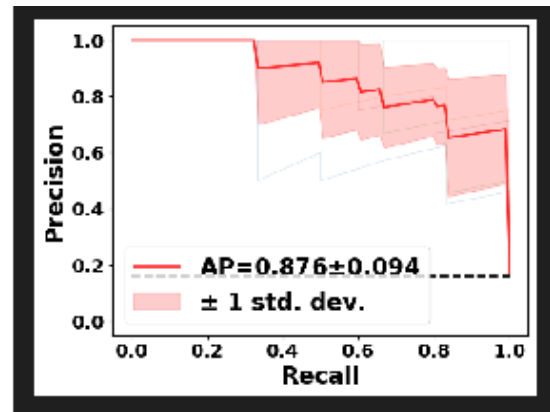
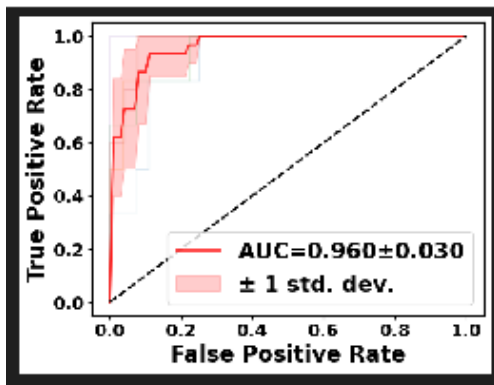
绘制ROC曲线及PR曲线

```

from performance_eval import generate_roc, generate_pr
generate_roc(
    ['tb_log/CNN_baseline_new_cross0/test_eval.csv', 'tb_log/CNN_baseline_new_cross1/test_eval.csv', 'tb_log/CNN_baseline_new_cross2/test_eval.csv', 'tb_log/CNN_baseline_new_cross3/test_eval.csv'],
    'ROC',
    'roc',
    'ADMI test roc')
generate_pr(
    ['tb_log/CNN_baseline_new_cross0/test_eval.csv', 'tb_log/CNN_baseline_new_cross1/test_eval.csv', 'tb_log/CNN_baseline_new_cross2/test_eval.csv', 'tb_log/CNN_baseline_new_cross3/test_eval.csv'],
    'PR',
    'pr',
    'ADMI test pr')

```





Shap可解释分析

```
shap_values, _ = model.shap("COG_shap_background4")
```

