README

Document Tree

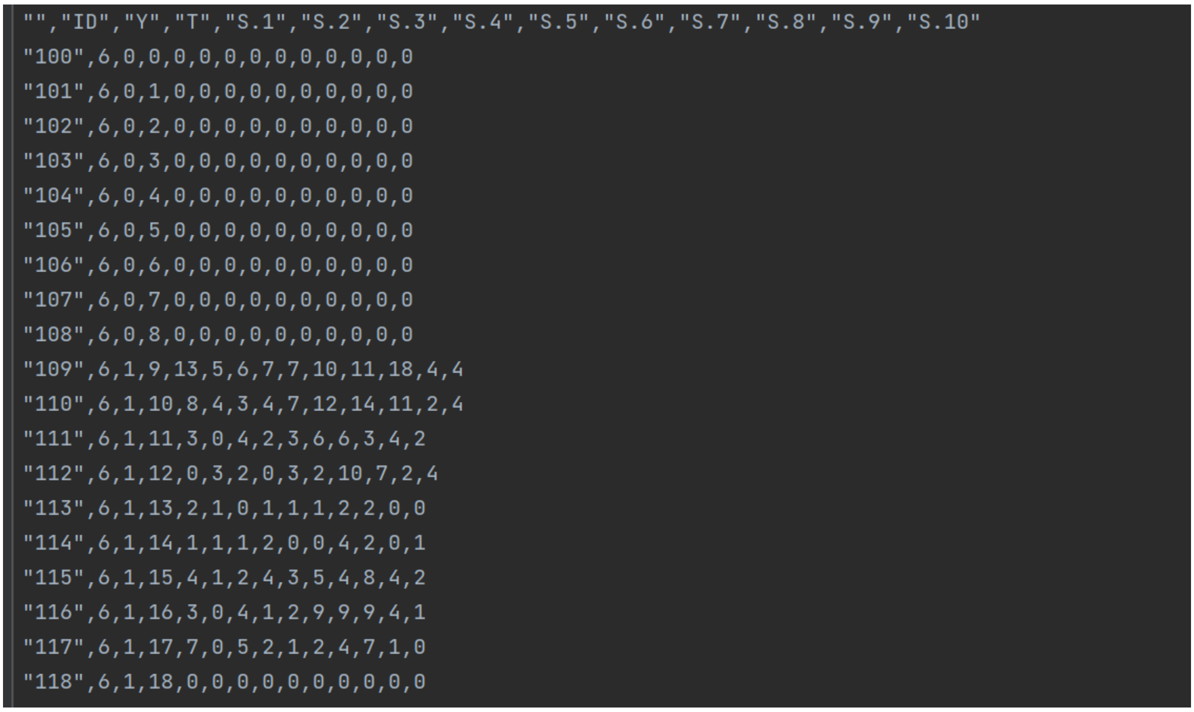
```
|  a_semi_model_final.py
|  a_train_final.py
|  a_train_final_sim.py                          # model running script
|  a_Transformer.py
|  a_utilize_semi.py
|  generate_Sigma.py                             # Sigma generating script
|  gen_Data_SIM.R                                # SimDat generating script
|  get_Folds.R
|  pack_prepare.R                                # package preparation for R
scripts
|  ReadMe_Simulation.txt
|  run_example.sh                                # Example for the whole
pipeline
|  Simdat_generation.pdf                         # Principle of simulation data
generation
|  simulation_data_functions_v4.R
|
├─.idea
|
├─data
|      embedding_selected.csv                    # embedding_selected for
simulation
|      embedding_selected.npy
|      Norms.csv
|      Sigma.csv                                 # sigma generated for SimDat
genration
|      Sigma.npy
|
├─example                                        # example for embeddings and
input data
|      embedding_example.csv
|      Input_example.csv
|      input_example_sim.csv
|
├─Results
├─Results_example
|
├─Simulation
```

```
|   └─SimDat                                  # Simdat generated by me
|       ├─SimDat.1                            # Simdat.1 is generated using
|       |  |  SimDat.1.csv                    # All of SimDat.1
|       |  |  SimDat.1.Rds
|       |  |  SimDat.1_labeled.csv            # labeled part of SimDat.1
|       |  |  SimDat.1_label_patient_info.csv
|       |  |  SimDat.1_unlabeled.csv          # unlabeled part of SimDat.1
|       |  |  SimDat.1_unlabeled.csv6116_ALL.pkl
|       |  |  SimDat.1_unlabeled.csv9061_ALL.pkl
|       |  |  test_patients.csv               # test patients' numbers
|       |  |  train_patients_450.csv          # train patients' numbers
|       |  |  unlabeled_patients.csv          # unlabeled patients' numbers
|       |  |
|       |  ├─test
|       |  |      test_data.csv
|       |  |
|       |  └─train                            # Simdat.2 is generated using
|       |          train_data.csv
|       |
|       └─SimDat.2
```

Input Data & Data Generation

Input Structure

- ID: patient_num
- Y: the label 0 or 1
- T: integers that flags the dates
- other columns for the covariant(columns length is flexible)



```
"","ID","Y","T","S.1","S.2","S.3","S.4","S.5","S.6","S.7","S.8","S.9","S.10"
"100",6,0,0,0,0,0,0,0,0,0,0,0,0
"101",6,0,1,0,0,0,0,0,0,0,0,0,0
"102",6,0,2,0,0,0,0,0,0,0,0,0,0
"103",6,0,3,0,0,0,0,0,0,0,0,0,0
"104",6,0,4,0,0,0,0,0,0,0,0,0,0
"105",6,0,5,0,0,0,0,0,0,0,0,0,0
"106",6,0,6,0,0,0,0,0,0,0,0,0,0
"107",6,0,7,0,0,0,0,0,0,0,0,0,0
"108",6,0,8,0,0,0,0,0,0,0,0,0,0
"109",6,1,9,13,5,6,7,7,10,11,18,4,4
"110",6,1,10,8,4,3,4,7,12,14,11,2,4
"111",6,1,11,3,0,4,2,3,6,6,3,4,2
"112",6,1,12,0,3,2,0,3,2,10,7,2,4
"113",6,1,13,2,1,0,1,1,1,2,2,0,0
"114",6,1,14,1,1,1,2,0,0,4,2,0,1
"115",6,1,15,4,1,2,4,3,5,4,8,4,2
"116",6,1,16,3,0,4,1,2,9,9,9,4,1
"117",6,1,17,7,0,5,2,1,2,4,7,1,0
"118",6,1,18,0,0,0,0,0,0,0,0,0,0
```

see example input at: `/example/input_example_sim.csv`

Data Generation

you need to download R and run `pack_prepare.R` before generating simulation data.

See `run_example.py` for an example of the **whole pipeline.**

Principle of simulation data generation.

See detailed description at `/Simdat_generation.pdf`

Pipeline of data generation

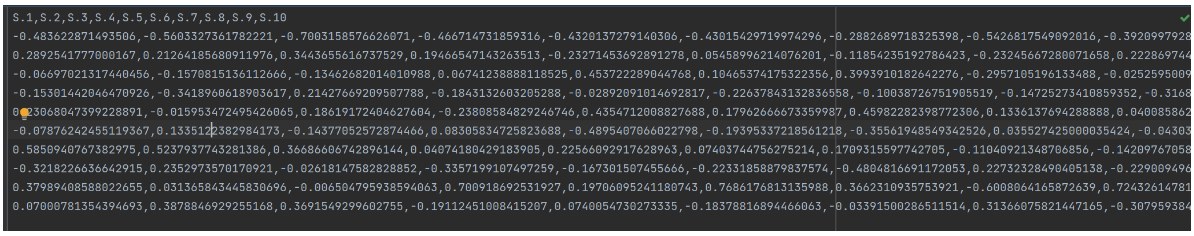
1. **First generate Sigma for multivariant norm distribution.**

run `genrate_Sigma.py`

this python script **generate simulation embeddings** by cutting example_embedding.csv. **Then generate Sigma matrix** using dot production of different concepts' embeddings.

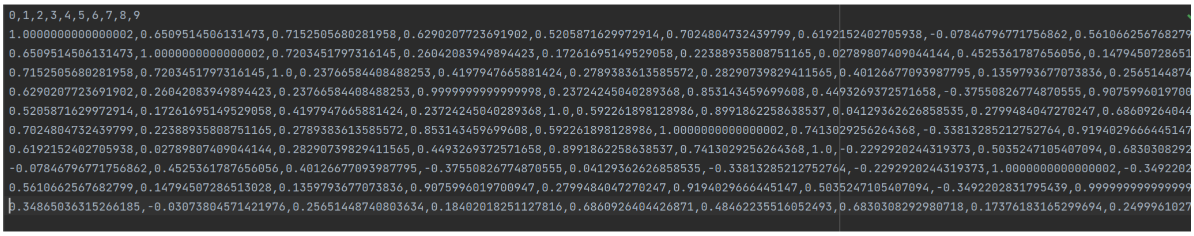
- selected embeddings saved at `/data/embeddings_selected.csv`
- Sigma saved at `` `/data/Sigma.csv ``

The **embedding dimension** and **covariant column number** is both set to 10 initially, change the `d` or `l` in the script if needed. Remember to **adjust the column names** at line.35 if changing the column numbers



```
S.1,S.2,S.3,S.4,S.5,S.6,S.7,S.8,S.9,S.10
-0.483622871493506,-0.5603327361782221,-0.7003158576626071,-0.466714731859316,-0.4320137279140306,-0.43015429719974296,-0.2882689718325398,-0.5426817549092016,-0.3920997928
0.289254177000167,0.21264185680911976,0.3443655616737529,0.19466547143263513,-0.23271453692891278,0.05458996214076201,-0.11854235192786423,-0.23245667280071658,0.222869744
-0.0669702131740456,-0.1570815136112666,-0.13462682014010988,0.06741238888118525,0.453722289044768,0.10465374175322356,0.3993910182642276,-0.2957105196133488,-0.0252595009
-0.15301442046470926,-0.3418960618903617,0.21427669209507788,-0.1843132603205288,-0.02892091014692817,-0.22637843132836558,-0.10038726751905519,-0.14725273410859352,-0.3168
0.23068047399228891,-0.015953472495426065,0.18619172404627604,-0.23808584829246746,0.4354712008827688,0.17962666673359987,0.45982282398772306,0.1336137694288888,0.040085862
-0.07876242455119367,0.1335122382984173,-0.14377052572874466,0.08305834725823688,-0.4895407066022798,-0.19395337218561218,-0.35561948549342526,0.035527425000035424,-0.04303
0.5850940767382975,0.5237937743281386,0.36686606742896144,0.04074180429183905,0.22566092917628963,0.07403744756275214,0.1709315597742705,-0.11040921348706856,-0.14209767058
-0.32182266366642915,0.2352973570170921,-0.02618147582828852,-0.3357199107497259,-0.167301507455666,-0.22331858879837574,-0.4804816691172053,0.22732328490405138,-0.229009496
0.37989408588022655,0.031365843445830696,-0.006504795938594063,0.700918692531927,0.19706095241180743,0.7686176813135988,0.3662310935753921,-0.6008064165872639,0.72432614781
0.07000781354394693,0.3878846929255168,0.3691549299602755,-0.19112451008415207,0.0740054730273335,-0.18378816894466063,-0.03391500286511514,0.31366075821447165,-0.307959384
```

Example of selected embeddings



```
0,1,2,3,4,5,6,7,8,9
1.0000000000000002,0.6509514506131473,0.7152505680281958,0.6290207723691902,0.5205871629972914,0.7024804732439799,0.6192152402705938,-0.07846796771756862,0.561066256768279
0.6509514506131473,1.0000000000000002,0.7203451797316145,0.26042083949894423,0.17261695149529058,0.22388935808751165,0.02789807409044144,0.4525361787656056,0.1479450728651
0.7152505680281958,0.7203451797316145,1.0,0.23766584408488253,0.4197947665881424,0.2789383613585572,0.28290739829411565,0.40126677093987795,0.1359793677073836,0.2565144874
0.6290207723691902,0.26042083949894423,0.23766584408488253,0.9999999999999998,0.2372424504028936 8,0.853143459699608,0.4493269372571658,-0.37550826774870555,0.9075996019700
0.5205871629972914,0.17261695149529058,0.4197947665881424,0.2372424504028936 8,1.0,0.592261898128986,0.8991862258638537,0.04129362626858535,0.2799484047270247,0.6860926404 4
0.7024804732439799,0.22388935808751165,0.2789383613585572,0.853143459699608,0.592261898128986,1.0000000000000002,0.7413029256264368,-0.33813285212752764,0.9194029666445147
0.6192152402705938,0.02789807409044144,0.28290739829411565,0.4493269372571658,0.8991862258638537,0.7413029256264368,1.0,-0.2292920244319373,0.5035247105407094,0.6830308292
-0.07846796771756862,0.4525361787656056,0.40126677093987795,-0.37550826774870555,0.04129362626858535,-0.33813285212752764,-0.2292920244319373,1.0000000000000002,-0.3492202
0.5610662567682799,0.14794507286513028,0.1359793677073836,0.9075996019700947,0.2799484047270247,0.9194029666445147,0.5035247105407094,-0.3492202831795439,0.9999999999999999
0.34865036315266185,-0.03073804571421976,0.25651448740803634,0.18402018251127816,0.6860926404426871,0.48462235516052493,0.6830308292980718,0.17376183165299694,0.2499961027
```

Example of generated Sigma

2. **Generate simulation data**

run `Rscript gen_Data_SIM.R 2000 500 100 200` at **terminal**

```
Rscript gen_Data_SIM.R {1} {2} {3} {4}
# ARGS: 1.total number 2.labeled number 3.train(labeled) num 4. test(labeled) num
#Rscript gen_Data_SIM.R 2000 500 100 200
```

Generated data is saved at `/Simulation/SimDat`

- SimDat.1 & SimDat.2 are generated using different Hazard Models.
- Remember to adjust the line.213-line.222 of `simulation_data_functions_v4.R` if changing the column numbers

Run Model

run `a_train_final_sim.py`

- Look into `Simulation_runmodel_descript.txt` for description of different ARGS.

Result Structure

- Results are saved in `/Results/`, you can see example results at `/Results_Example/`
- Document Tree of Results directory:

```
C:.
|   Incident_epoch29_818__results_RETTAIN.csv      # important
|   Incident_epoch30_818__results_RETTAIN.csv      # important
|   Prevalence__818__results_RETTAIN.csv           # important
|
results_RETTAIN.csv1_Attenation_value_patient_visit_code_prediction_label_weight_test.pkl
|
results_RETTAIN.csv1__embedding_patient_ori_codeRW_codeVistRW_hiddenFCN_label_test.pkl
|   results_RETTAIN.csv1__embedding_patient_ori_hiddenFCN_label_train.pkl
|   results_RETTAIN.csv_incident_evaluation.txt    # important
|   results_RETTAIN.csv_prevalence_evaluation.txt  # important
|   _results_RETTAIN.csv_code_weights.csv
|
└─results_RETTAIN.csv_model
    |   keras_metadata.pb
    |   saved_model.pb
    |
    ├─assets
    └─variables
            variables.data-00000-of-00001
            variables.index
```

  - Incident_epoch{}__results_RETTAIN.csv : **incident phenotyping(prediction on every time spot) results**
  - Prevalence__results_RETTAIN.csv: **Prevalence(EVER/NEVER) binary phenotyping results**
  - .txt: the calculated AUC, ACC, F1, Speci etc.