User Manual

1 Summary

Time series gene expression in TEDDY is collected for a subset of enrolled participants and even those participants have huge amount of data missing. This framework is designed to impute gene expression for all participants, whether they have partially or completely missing gene expression. The synthetic gene expression is integrated with other risk factors to improve the prediction of islet autoimmunity and illustrate the positive effect this computation approach provides.

2 Download

The framework is downloadable directly from github. Users need to have python (version 3.0 or higher) installed in their machine.

3 Required python packages

- Numpy (>=1.17.2)
- Pandas (>=0.25.1)
- sklearn (>=0.21.3)
- PyTorch (pytorch version >=1.5.0, torchvision version >=0.6.0)

4 Running the framework

Imputation

Datasets first need to be processed and imputed before we can perform the prediction.

data_processing.py: Run *data_processing.py* to pre-preprocess the gene expression and SNP data for imputation. It converts the gene expression into a three dimensional time series data between 3-48 months, finds top 50 principal components of SNP, matches samples in gene expression and SNP.
$ python data_processing.py --data_directory /home/tanvir/Diabetes/data/raw_data/ --save_directory /home/tanvir/Diabetes/data/processed_data/

--data_directory: directory of raw omics data
--save_directory: directory to save processed data

imputation.py: Run *imputation.py* to perform gene expression imputation on processed data.
$ Python imputation.py --data_directory /home/tanvir/Diabetes/data/processed_data/ --save_directory /home/tanvir/Diabetes/data/imputed_data/ --n_epochs 100 --batch_size 32 --learning_rate 1e-3 -- true_enc_hidden_size 100 --true_enc_batch_size 32 --true_enc_num_epochs

100 -- true_enc_learning_rate 1e-4 --true_enc_gamma 0.99 --syn_enc_num_epochs 25
--syn_enc_batch_size 512 --syn_enc_learning_rate 1e-5 --data_directory: directory of input data

--save_directory: directory to save output
--n_epochs: number of epoch for C_1
--batch_size: batch size used in C_1
--learning_rate: learning rate used in C_1
--true_enc_hidden_size: size of the encoding in C_0
--true_enc_batch_size: batch size used in C_0
--true_enc_num_epochs: number of epoch for C_0
--true_enc_learning_rate: learning rate used in C_0
--true_enc_gamma: *gamma* value in learning rate scheduler for C_0
--syn_enc_num_epochs: number of epoch for C_2
--syn_enc_batch_size: batch size used in C_2
--syn_enc_learning_rate: learning rate used in C_2

Prediction

prediction.py: Run prediction.py to predicts IA status of participants using imputated data.
$ python prediction.py --hidden_size 200 --num_layers 3 --num_epochs 5 --batch_size 8 --learning_rate 0.00001 --end 24 --serial 16 --option 0 --imputed_data_dir /home/tanvir/Diabetes/data/imputed_data/ - -processed_data_dir /home/tanvir/Diabetes/data/processed_data/
--raw_data_dir /home/tanvir/Diabetes/data/raw_data/

--hidden_size: size of the hidden representation in LSTM
--num_layers: number of layers in LSTM
--num_epochs: number of training epochs in LSTM
--batch_size: batch size in LSTM
--learning_rate: learning rate in LSTM
--end: IA time cutoff in months
--serial: gene expression cutoff in number of time steps
--option: run the model using combined data or only gene expression. Use 0 for combined data and 6 for only gene expression.
--imputed_data_dir: save directory for imputation.py
--processed_data_dir: save directory for data_processing.py
--raw_data_dir: directory of raw omics data

5 Datasets

Gene expression and SNP can be downloaded from this link: `https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs001442.v1.p1`

6 Example run

As the TEDDY datasets are protected, we provide dummy datasets to show the workflow of the proposed framework. Dummy SNP and gene expression datasets can be downloaded from this link
*prediction*1*.py* is a simplified version of our prediction algorithm that shows the workflow using only imputed gene expression and SNPs. To go through the actual prediction algorithm please refer to *prediction.py* which can only be run with TEDDY datasets. The study is designed to solve the limitation of missing values in TEDDY datasets to predict IA. For a more generalized

approach involving multi-modal time series and cross-sectional datasets, we have developed another framework downloadable from here.