# **Software Description**

# Overview

This is the software script and data of our deep transfer learning method for GWAS (Genome wide Association Study) published on XXX, as well as reproducing the results described in the manuscript.

# Software Structure

This software contains three major components:

* data: it contains the instructions for downloading the 14 synthetic datasets used in our experiments, as well as the SNPs and clinical variables used in each disease study.
* examples: this directory contains the scripts to generate the data shown in the four plots of Fig 2 and Table 1.
* simulation1: this directory contains the scripts used for 8 synthetic datasets, to generate the plots shown in Fig 4 and Table 2.
* simulation2: this directory contains the python scripts used to generate the plots in Fig S1 and Table S2.

| **Entity** | **Path/location** | **Note** |
| --- | --- | --- |
| Data | ./data/instructions.txt | The path of the synthetic datasets |
| Features | ./data/Features.xlsx | The features used for each study |
| Model generator | ./\*.py/build\_model | The interface used to build a deep learning model |
| Mixture | ./examples/\*.py/mixture\_learning | The mixture learning scheme for each study. |
| Independent | ./examples/\*.py/independent\_learning | The independent learning scheme for each study |
| Naïve Transfer | ./examples/\*.py/naive\_transfer | The Naïve transfer learning scheme for each study |
| Transfer | ./examples/\*.py/super\_transfer | The transfer learning scheme for each study |

# System Requirements

## Software dependency

The system relies on the following software, reagent, or resources.

## Software version

Our software has been tested on the following software version.

| **Software and Hardware** | **SOURCE** | **IDENTIFIER** |
| --- | --- | --- |
| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
| Python 3.7 | Python Software Foundation | <https://www.python.org/download/releases/2.7/> |
| Computational Facility | The National Institute for Computational Sciences | <https://www.nics.tennessee.edu/computing-resources/acf> |
| Numpy 1.15.4 | Tidelift, Inc | <https://libraries.io/pypi/numpy/1.15.4> |
| Numpydoc 0.9.1 | Tidelift, Inc | <https://libraries.io/pypi/numpydoc> |
| Scipy 1.2.1 | The SciPy community | <https://docs.scipy.org/doc/scipy-1.2.1/reference/> |
| Sklearn 0.0 | The Python community | <https://pypi.org/project/sklearn/> |
| Keras 2.2.4 | GitHub, Inc. | <https://github.com/keras-team/keras/releases/tag/2.2.4> |
| Keras-Applications 1.0.8 | GitHub, Inc. | <https://github.com/keras-team/keras-applications> |
| Keras-Preprocessing 1.1.0 | GitHub, Inc. | <https://github.com/keras-team/keras-preprocessing/releases/tag/1.1.0> |
| Tensorboard 1.13.1 | GitHub, Inc. | <https://github.com/tensorflow/tensorboard/releases/tag/1.13.1> |
| Tensorflow 1.13.1 | tensorflow.org | <https://www.tensorflow.org/install/pip> |
| Tensorflow-estimator 1.13.1 | The Python community | <https://pypi.org/project/tensorflow-estimator/> |
| Statsmodels 0.9.0 | Statsmodels.org | <https://www.statsmodels.org/stable/release/version0.9.html> |
| Xlrd 1.2.0 | The Python community | <https://pypi.org/project/xlrd/> |
| XlsxWriter 1.1.8 | The Python community | <https://pypi.org/project/XlsxWriter/> |
| Xlwings 0.15.8 | The Python community | <https://pypi.org/project/xlwings/> |
| Xlwt 1.3.0 | The Python community | <https://pypi.org/project/xlwt/> |

## Hardware requirements

We recommend using a GPU (V100) to speed up the running process of our software.

# Installation Guide

Our software package can be downloaded from the following Github homepage: https://github.com/AtlasGao/GWAS\_DTL. This package contains the source code and instructions to reproduce the results represented in our paper. Our software would run on Windows and Ubuntu, but we suggest using Linux system which is easier for environment configuration.

Conda –install requirements.txt

Requirements.txt

numpy==1.15.4

numpydoc==0.9.1

scipy==1.2.1

seaborn==0.9.0

sklearn==0.0

skrebate==0.6

torch==1.7.1

Keras==2.2.4

Keras-Applications==1.0.8

Keras-Preprocessing==1.1.0

tensorboard==1.13.1

tensorflow==1.13.1

tensorflow-estimator==1.13.0

statsmodels==0.9.0

lifelines==0.16.3

Optunity==1.1.1

xlrd==1.2.0-

XlsxWriter==1.1.8

xlwings==0.15.8

xlwt==1.3.0

# Demo

## Instructions to run each experiment

The python scripts used to generate the Lung cancer and the Prostate cancer in our paper can be found in the following folder:

cd /GWAS\_DTL/diseases

python Lung\_cancer\_European\_EastAsian.py

python Prostate\_cancer\_European\_AfricanAmerican.py

The python scripts used to generate the data of Table 2 and Figure 4 in our paper can be found in the following folder

cd / GWAS\_DTL / simulation1

python \*.py

The python scripts used to generate the data of Table S2 and Figure S1 in our paper can be found in the following folder

cd / GWAS\_DTL / simulation2

python \*.py

After the execution, the result will be printed in the console.

## Expected output

The output of each script will be a data frame with 7 columns and 20 rows. Each row represents a single run, and the 7 columns show the result of different machine learning schemes.

# Instructions for Use

## How to run the software

To run our software with different diseases, you need to download our dataset from the FigureShare server and put it under the GWAS\_DTL/data/ folder. You can simply specify the task you want to run by redirecting to the location of the script of a specific task.

## Reproduction instructions

The key point to reproduce the result in our paper is to follow the configuration process strictly.

## Authors

Yan Gao and Yan Cui, ({ygao45, ycui2}@uthsc.edu).

## License

This project is covered under the GNU General Public License (GPL).