Research Goal: Integrating ChaNGa with Machine Learning Algorithms

- **ChaNGa:**
  - runs detailed, physics-based simulations of the astrophysical scenario under consideration
  - produces simulation data
  - provides the training dataset for machine learning algorithms
- **Machine Learning Algorithms:**
  - learning patterns, correlations, and features from the simulated data
  - the trained model from a machine learning algorithm provides a data-driven, predictive tool that can make estimates or predictions beyond original simulation data
  - **One choice: Support Vector Regression (SVR)**
    - good for predicting the position of stars based on their previous positions and velocities
    - easy to implement
    - could handle large datasets with high dimensionality

Installing ChaNGa:

1. download Charm: git clone https://github.com/UIUC-PPL/charm
2. cd charm
3. git checkout v7.0.0
4. build Charm(successful): ./build charm++ netlrts-darwin-x86_64 smp --with-production -j4
5. ../
6. git clone utility
7. cd changa
8. ./configure
9. make

Useful ChaNGa Commands:

- cd changa
- specify input file—requires an input file to specify the parameters: ./changa -i filename.txt
- specify output directory—specify a directory where ChaNGa should write its output files: ./changa -o directory
- set a specific parameter: ./changa -p parameter_name=parameter_value
- debug mode: ./changa -v
- run in Parallel—use -n to specify the number of processes or threads: ./changa -n 4

Running Data:

1. ssh
2. vim changa.sh
3. sbatch changa.sh (compile)
4. ls (check which one is the output file)
5. vim slurm-XXXX.out (open the output file)

File List:

- **dwf1.2048.00384**: binary file of input data
- **decoding_tipsy_input.py**: file to decode the binary file dwf1.2048.00384
- **decoding_tipsy_input.txt**: the output of decoding_tipsy_input.py
- **dwf1.2048.bench.000001—dwf1.2048.bench.000010**: binary file of output data for each step (total 10 steps)
- **decoding_tipsy_all.py**: file to decode the binary files dwf1.2048.bench.000001-dwf1.2048.bench.000010
- **decoding_tipsy_all_1.txt—decoding_tipsy_all_3.txt**: the output of decoding_tipsy_all.py
- SVR.py: file for applying Support Vector Regression model

File Details & Challenges:

- **decoding_tipsy_input.py**: generate input data for SVR model.
  - Code following instructions from https://github.com/N-BodyShop/changa/wiki/File-Formats
  - Seems only have dark particles(no gas particles and star particles)
  - Output file: decoding_tipsy_input.txt
- **decoding_tipsy_all.py**: generate output data for SVR model.
  - Code following instructions from https://github.com/N-BodyShop/changa/wiki/File-Formats
  - Seems only have dark particles(no gas particles and star particles)
  - I experimented with three files dwf1.2048.bench.000001—— dwf1.2048.bench.000003 and got output files decoding_tipsy_all_1.txt——decoding_tipsy_all_3.txt.
  - Next step: run all 10 files
- **SVR.py**: Support Vector Regression model
  - This file still produces errors (mainly about N/A values and data dimension) and needs further debugging
    - ValueError: y should be a 1d array, got an array of shape (3981158, 3) instead