README

Packages

DataReader: A package for reading data.

Parameters:

num: the total number of data points.
data_path: the path of data file.
target_path: the path of target file.

Methods:

read_data_point(): read a data point and return it.
read_data_chunk(chunk_size): read chunk_size data points and return them.

online_ensemble: A general ensemble package. It can do basic prediction. And it also supports dynamically inserting or deleting regressors/classifiers.

Parameters:

n_jobs: the number of jobs to run in parallel for training new estimators.

Attributes:

_estimators: a dictionary of estimators in the ensemble. The key is the unique identifier.
_cnt: the number of trained estimators in total.

Methods:

insert(n_estimators, x, y, category): n_estimators is the number of estimators to be inserted. (x, y) is a data chunk for training. category is the category of estimators to be inserted.
insert_with_SVM_regressor(n_estimators, x, y): insert n_estimators SVM regressors using (x, y).
insert_with_random_forest_regressor(n_estimators, x, y): insert n_estimators RF regressors using (x, y).
insert_with_random_forest_classifier(n_estimators, x, y): insert n_estimators RF classifiers using (x, y).
insert_with_estimators(estimators): insert external specified estimators.
delete(idx_list): delete specified estimators according the index list.
get_idx_list: return the index list of all estimators.
predict_results(x): return results from all estimators.
predict_weighted_sum(x, weights): return a weighted sum regression result.
predict_weighted_classification_result(x, weights): classification result from the weighted sum regression result.
predict_weighted_vote(x, weights): the most voted classification result.

Programs for regression

anil_master_fast_convergence.py: the last version of Anil's algorithm for concept drift.

Methods:

test(n_jobs, n_trees, category, chunk_size)
accept: n_jobs is the number of job to run in parallel. n_trees is the number of estimators to build in the first. category is the category of estimator. chunk_size is the size of data chunk used for training. Please set directory, data_file_name, target_file_name to open data files.
return: mse vector, replacement vector, final mse, execution time

Programs for classification

gen.py: following Wang's paper, this file can generate a data set for testing. You can set several parameters to specify the data you want to generate.

master_classification.py: Anil's algorithm for classification

wang_algo.py: Wang's algorithm for classification

single.py: Singe estimator algorithm for classification

Methods:

all 3 classification program have the same method called run_classification.

accept: data and target file names, the number of chunks to be tested, the size of chunk and the number of classifiers before calling it.

return: final error rate, error_rate_vec

Programs for testing

test.py: it can test different algorithms and present their plots into one plot.

Results of classification experiments

These files are too big to put on github. You can acces them via https://drive.google.com/a/nyu.edu/folderview?id=0B_JtB81Yrq-LfjNvMTlwSEZkbWphd1JCRTVJNHhsQ2xqR2M1TGFXTUZaVk9HSHlzSHNiVGc&usp=sharing

Results of regression experiments

These files are put into result directory on github. png files are all plots and result_all.txt records the execution time for all experiments.