Regression models for predicting the efficiency of organic photovoltaic:

The goal of this assignment was to predict an optical feature for many kinds of materials which indicate their potentials as solar cells. We built models with features extracted from 1 million molecular structures. A chemistry package in python can produce good feature representation. And we attempted to plug those engineered molecular properties into linear regression models, adaboost, gradient boosting and random forest. Linear models were regularized by lasso and ridge shrinkage whose tuning parameters were selected based on 5 fold cross validation. The parameters of adaboost and gradient boosting and random forest were also varied to optimize the models. We came to a conclusion that for this specific problem random forest has the best performance among all models attempted.(mean square error, 512 features)

Multilabel classification on malicious software:

We classified executable files collected from people’s computers into any of 14 known malware classes or determine that the executables are not malware. The evaluation metric is the percentage of correctly classified samples. (categorization accuracy). We tried random forest, neural network and SVM (with string kernels) using over 300 extracted feature. In terms of feature engineering, string subsequence kernel was utilized to address the inherent correlation in the executables. Neural network did not meet our expectation probably due to insufficient samples. With only several thousand samples it is hard to train a complex network structure. Once more data can be accessed we would expect that neural network can precede random forest.

Sound classification:

1. we focused on developing machine learning methods to identify the types of sounds using their amplitudes series over time. The original input was the amplitude series of audio clips. We first extracted acoustic features from these series using an open source library called librosa. The extracted features include Chromagram, Tonal centroid features,

By doing so, we avoided fitting a bunch of features highly correlated in time to classifiers. After that we performed PCA on the extracted features so as to alleviate multicollinearity and to exclude redundant features. With these transformed features we built logistic regression models, random forest, SVM and neural network as well as tuned parameter for each type of model to optimize their performance.

**Training of a reinforcement learning** agent for a game similar to super mario.

We abstracted the game setup to the definition of states, rewards, actions and policies in a Markov decision process. We implemented three reinforcement learning algorithms tabular SARSA, tabular Q-learning and deep Q-learning. Within SARSA and Q-learning we took a variant of epsilon greedy strategy called epsilon-decay which gradually decreases epsilon as learning goes on.