Project Proposal

Hyperspectral Data Classification

Team members names
Purdue University
wang1766@purdue.edu, yyy@purdue.edu, zzz@purdue.edu

October 1, 2015

1 Introduction

What is the project

Motivation

2 Related work

Describe any previous work you found related to your project

3 Problem formulation

Describe your project as a machine learning problem, identify inputs objects, labels, possible features

4 Data and Evaluation plan

The data set we plan to use is from a hyperspectral image that was taken by a drone. The whole data set contains all the pixels within the image, which will be further divided into training set and testing set. The whole data set contains 21025 rows, 200 columns of features and 1 column of label. All the feature values were normalized and ranged from 0 to 1000. The label column contains 16 different values. In other words, we have a multi-classification problem.

Below is a pilot evaluation plan:

- Data Check
  The validity of the whole data set will be checked. Any missing values, non-numeric values and outliers will be identified.

- Data split
  The whole data set will be split into training and testing set. Because the data set is unbalanced (the smallest label group contains 20 examples, while the largest label group contains more than 2000 examples), up-sampling of small label groups or down-sampling of the large label groups will be conducted when the training data is selected. In addition, the training set will be further randomly divided into 5 folds of equal size, in order to facilitate the following cross validation process.

- Feature Selection
  Considering the dimensionality of the data set is relatively high, Principal component analysis (PCA) will be used to conduct dimensionality reduction. We plan to keep at least 90% of the original variance, and the principal components will be used in the following data analysis.

- Model Selection
  Considering our task is classification based on numeric features, we plan to construct models using the following algorithms:

  - Random Forest
    Hyper-parameter to be tuned: Randomly selected features, maxiumn number of trees. This model is available in the Sklearn package of Python.
  - Neural Network
    For this model, BackpropTrainer algorithm from Pybrain package of Python will be used. The hyper-parameters include: the hidden layers, learning rate, batch size and so on.
  - Linear SVM For this model, the tunable hyper-parameters include cost and maximum iterations. This model is available in the Sklearn package of Python.
  - Kernel SVM If necessary, different kernel functions will test. The available kernels in the Sklearn package include: linear, polynominal, rbf, sigmoid and precomputed.

- Model evaluation
  After the construction of the above models, the performance of the different models will be evaluated by using 5-fold cross validation. Since the correct label of all data points are known, misclassification rate will be used as the subjective function.

- Model testing
  For the models with satisfied performance, the error rate will be tested against the held out testing data.