

Introduction to Supervised Learning: Homework 2*Professor: Greg Grudic; Substitute Teacher: Sam Reid*

For additional instructions, please visit the website at:
<http://www.colorado.edu/physics/pion/csci5622-spring08/>

Problem 1

What is the loss function for the Netflix Prize?

Problem 2

What algorithm is used in the highest-scoring Netflix Prize submission? Give a high-level description; answer in 50 words or less.

Problem 3

A machine learning researcher gives you a classifier for a particular problem that you evaluate to have 97% accuracy over unseen data. How much better is it possible to do on this problem?

Problem 4

a) Implement stratified data set partitioning for binary classification data in Matlab®. This function should have the following signature:

```
function partitions=partition(fullDataset,partitionSizes)
```

where:

- *fullDataset* is the entire data set to be partitioned. The data set is a matrix in which a data point is a row, and the target class (0 or 1) is in the rightmost column.
- *partitionSizes* is a list (row matrix) of partition sizes
- *partitions* is a cell array of data sets.

b) Implement the K-Nearest Neighbor algorithm for binary classification. This can be a modified version of your homework submission from last week. The signature should be the following:

```
function [knnData functionHandle]=buildKNNModel(trainSet,k)
```

where

- *trainSet* is the training set matrix, where each row is a data point and the class value (0 or 1) is in the last column
- *k* is the hyperparameter “k” for the KNN algorithm; the number of neighbors to use
- *knnData* is any data KNN needs during classification
- *functionHandle* is the function used to evaluate the trained KNN model, as in last week’s homework. This function should take the *knnData* and unlabeled points and return their labels.

c) Write a cross-validation algorithm in Matlab®. The signature should be the following:

```
function crossValidationLosses=crossValidate(dataset,algorithms,hyperparameters,numFoldsCV)
```

where:

- *dataset* is a matrix in which each row is a data point and class values are in the rightmost column.
- *algorithms* is a list (row matrix) of function handles for classification algorithms, as in last week’s homework
- *hyperparameters* is a list (row matrix) of hyperparameters one for each classification algorithm
- *numFoldsCV* is the number of folds to be used for cross validation
- *crossValidationLosses* is a list (row matrix) of the cross-validation scores (loss function averaged over all folds), one for each algorithm

Notes:

1. Partition the dataset into stratified cross-validation folds.
2. Use the Zero-One loss function $L = \frac{1}{N} \sum_{i=1}^N 1 - \delta(y_i - \hat{y}_i)$
3. Each model should be cross-validated on the same partitions.
4. Please see `main.m` on the website for sample usage of these functions.

For formatting and submission instructions, please see the webpage at:
<http://www.colorado.edu/physics/pion/csci5622-spring08/>