# ECE523: Engineering Applications of Machine Learning and Data Analytics
# Due 03/27/2017 @ 11:59PM (D2L)

**Name**: _____

**Signature**: _____

**Date**: _____

**Instructions**: There are seven problems. Partial credit is given for answers that are partially correct. No credit is given for answers that are wrong or illegible. All work must be supported and code must be submitted for credit.

Theory: _____

Practice: _____

Total: _____

# Part A: Theory (20pts)

## (15pts) Support Vector Machines (Revisited)

We now look at a different type of SVM that is designed for domain adaptation and optimizes the hyperplanes given by $\mathbf{w}_S$ (source hyperplane) before optimizing $\mathbf{w}_T$ (target hyperplane). The process begins by training a support vector machine on source data then once data from the target are available, train a new SVM using the hyperplane from the first SVM and the data from the target to solve for a new "domain adaptation" SVM.

The primal optimization problem is given by

$$\begin{aligned} \min_{\mathbf{w}_T,\xi} \quad & \frac{1}{2}\|\mathbf{w}_T\|^2 + C\sum_{i=1}^{n}\xi_i - B\mathbf{w}_T^T\mathbf{w}_S \\ \text{s.t.} \quad & y_i(\mathbf{w}_T^T\mathbf{x}_i + b) \geq 1 - \xi_i && \forall i \in \{1,\ldots,n\} \\ & \xi_i \geq 0 && \forall i \in \{1,\ldots,n\} \end{aligned}$$

where $\mathbf{w}_S$ is hyperplane trained on the source data (*assumed to be known*), $\mathbf{w}_T$ is hyperplane for the target, $y_i \in \{\pm 1\}$ is the label for instance $\mathbf{x}_i$, $C$ & $B$ are regularization parameters defined by the user and $\xi_i$ is a slack variable for instance $\mathbf{x}_i$. The problem becomes finding a hyperplane, $\mathbf{w}_T$, than minimizes the above objective function subject to the constraints. Solve/derive the dual optimization problem.

## (5pts) AdaBoost: How much do you understand the algorithm?

In a couple paragraphs, answer: Why does boosting (generally) work?

# Part B: Practice (20pts)

You are free to use functions already implemented in Matlab, Python or R with the exception of problem 1. I recommend using Python's Scikit-learn (`http://scikit-learn.org/stable/`) as is implements most of the methods we will be discussing in this course... as well as problems in this homework!

## (15pts) Support Vector Machines (Revisited)

Implement the domain adaptation SVM from the first problem in the theory section. A data set for the source and target domains (both training and testing) have been uploaded to D2L. There are several ways to implement this algorithm. If I were doing this for an assignment, I would implement the SVM directly using quadratic programming. For example, see CVX (Python and Matlab: `https://goo.gl/3f7StQ`), but there are other tools available.

## (5pts) An Experiment with Ensembles

Perform an experiment that evaluates the impact of the size of an ensemble on the testing error of the `ionosphere.csv` data set. Sweep the ensemble size from 2 to whatever you want to

see of the testing error converges as the ensemble size is increased. Perform this experiment with Bagging and AdaBoost. Plot the testing error for both Bagging and AdaBoost in the same plot.