9.520 Problem Set 1
Due date: October 14, 2013

Note: there are five problems total in this set.

Problem 1 One common preprocessing in machine learning is to center the data. In this problem we will see how this can be related to working with an (unpenalized) offset term in the solution. Consider the usual Tikhonov regularization with a linear kernel, but assume that there is an unpenalized offset term \( b \),

\[
\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \langle w, x_i \rangle + b - y_i \right)^2 + \lambda \| w \|^2 \right\}
\]

and let \((w^*, b^*)\) be the solution of the above problem.

Denote by \( x_i^c = x_i - \bar{x} \), \( y_i^c = y_i - \bar{y} \) the centered data for \( i = 1, \ldots, n \), where \( \bar{y}, \bar{x} \) are the output and input means respectively. Show that \( w^* \) also solves

\[
\min_{w \in \mathbb{R}^d} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \langle w, x_i^c \rangle - y_i^c \right)^2 + \lambda \| w \|^2 \right\}.
\]

and determine the corresponding \( b^* \).

Problem 2 In classification problems where the data are imbalanced (i.e., there are many more examples of one class than of the other) a common strategy is to weight the loss function so that the errors in one class are counted more than those of the other class. In the case of RLS, this corresponds to solving the modified problem

\[
\min_{w \in \mathbb{R}^d} \left\{ \sum_{i=1}^{n} \gamma_i \left( \langle w, x_i \rangle - y_i \right)^2 + \lambda \| w \|^2 \right\}
\]

where \( \{\gamma_i : i = 1, \ldots, n\} \) are the weights of a convex combination (that is, \( \sum_{i=1}^{n} \gamma_i = 1 \) and \( \gamma_i > 0 \) for all \( i = 1, \ldots, n \)).

(a) Derive the explicit form of the minimizer \( w^* \) of the above problem.

(b) Consider the case of a weighted loss function and an offset \( b \in \mathbb{R} \),

\[
\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \left\{ \sum_{i=1}^{n} \gamma_i \left( \langle w, x_i \rangle + b - y_i \right)^2 + \lambda \| w \|^2 \right\}
\]

Using the above and the result from Problem 1, derive the explicit form of the minimizers \( w^*, b^* \) for this problem.

(c) What do you think could be a good (optimal?) way to choose the weights?

Note: This last question is somewhat open-ended. There is not necessarily a right or wrong answer. We are mostly interested in seeing your ideas and rationale.
Problem 3 The distance between two elements $\Phi(x), \Phi(x')$ of a feature space induced by some kernel $K$ can be seen as a new distance $d_K(x, x')$ in the input space.

(a) Show that such a distance can always be calculated without knowing the explicit form of the feature map itself.

(b) Consider a dataset of pairs $\{(x_i, y_i)\}_{i=1}^N$, with $x_i \in X$ and $y_i \in \{-1, 1\}$. Assume that $n_+, n_-$ of the $x_i$ have label $+1, -1$, respectively (so $n_+ + n_- = N$), and let’s also assume that we are given a kernel $K$ and an associated feature map $\Phi : X \to \mathcal{F}$ satisfying

$$K(x, x') = \langle \Phi(x), \Phi(x') \rangle_\mathcal{F}.$$ 

Derive a classification rule, involving only kernel products (and the sign function), that assigns to a new test point the label of the class whose mean is closest in the feature space according to the distance $d_K$.

Problem 4 In (binary) classification problems one aims at finding a classification rule (also called the “decision rule”) which is a binary valued function on the input space $c : X \to \{1, -1\}$. The quality of a classification rule can be naturally measured by means of the so called misclassification error

$$R(c) = \mathbb{P}\{c(x) \neq y\}.$$ 

If we introduce the misclassification loss $V(c(x), y) = \theta(-yc(x))$, where $\theta(s) = 1$ if $s > 0$ and $\theta(s) = 0$ otherwise, the misclassification error can be rewritten as

$$R(c) = \int_{X \times Y} \theta(-yc(x))p(x)p(y|x)dxdy.$$ 

Direct minimization of the misclassification error is not computationally feasible mostly because the misclassification loss is not convex. In practice, one usually looks for real valued (rather than binary valued) functions $f : X \to \mathbb{R}$ and replaces $\theta(-yc(x))$ with some convex loss $V(f(x), y) = \mathcal{L}(-yf(x))$ with $\mathcal{L} : \mathbb{R} \to \mathbb{R}$. A classification rule is then obtained by taking the sign, that is $c(x) = \text{sign}(f(x))$. Commonly chosen loss functions are the hinge loss and square loss. In this case the error is measured by the expected error

$$I[f] = \int_{X \times Y} \mathcal{L}(-yf(x))p(x)p(y|x)dxdy.$$ 

However, there is still the problem of relating the convex approximation to the original classification problem. With the above discussion in mind, and assuming that the distribution $p(x, y)$ is known, answer the following questions:

(a) Check that the square loss $V(f(x), y) = \|f(x) - y\|^2$ can be written in the form $V(f(x), y) = \mathcal{L}(-yf(x))$ for some function $\mathcal{L}$. Calculate the explicit form of the minimizer of $I[f]$ when $V$ is the square loss.

(b) Calculate the closed-form of the minimizer of $I[f]$ for $\mathcal{L}$ the exponential loss $\mathcal{L}(-yf(x)) = \exp(-yf(x))$.

(c) Find the closed-form of the minimizer of $I[f]$ with $\mathcal{L}$ the logistic loss $\mathcal{L}(-yf(x)) = \log(1 + \exp(-yf(x)))$.

(d) The minimizer of $R(c)$ over all possible decision rules is the so called Bayes decision rule $b : X \to \{1, -1\}$. For all the above considered losses, what is their relation to the Bayes decision rule?
Problem 5 (MATLAB).

In this exercise you will use regularized least squares (RLS) on a binary classification problem using artificial (synthetically generated) data.

Data: Download ps1-dataset.mat from the course web-page. It contains a training set Xtrain, Ytrain and a test set Xtest, Ytest, stored as $N \times d$ matrices, where $N = 100$ and $d = 2$. Both Xtrain and Xtest sets are 100 samples of two-dimensional features.

MATLAB library: GURLS which stands for Grand Unified Regularized Least Squares is a software library for regression and (multiclass) classification based on Regularized Least Squares (RLS). You will be using the GURLS module, implemented in MATLAB for the main functions (kernel computation, learning, prediction, regularization parameter selection) required in this exercise. GURLS is available through https://github.com/LCSL/GURLS. The instructions in Sec. 1.2.2 and 1.2.5 and the Examples in Sec. 2.2 and 2.3 of the Documentation are a great starting point for using the library. More info: http://lcsl.mit.edu/gurls.html

(a) [don't use GURLS] Implement a linear classifier using the primal formulation of RLS, using $K$-fold cross-validation (KFCV) to choose the regularization parameter $\lambda$. In $K$-fold cross-validation, the training set is partitioned into $K$ subsets, with each fold consisting of a split between $K - 1$ used for training and 1 used for validation. KFCV is often used when the amount of available data is small, so that one wants to use the entire dataset for both training and validation. Here you will use KFCV to select $\lambda$:

- Report the optimum $\lambda$ and the performance of RLS on the training set and test set with this $\lambda$. (Report the accuracy: the percentage of points correctly classified.)
- Plot the $K$-fold error vs. $\lambda$.

(Note: You will have to specify a value for $K$ and a grid for the tested values of $\lambda$. Reasonable values for $\lambda$ might range up to the maximum eigenvalue of the kernel matrix.)

(b) [use GURLS]: Train a linear classifier using the dual formulation (linear kernel) of RLS, choosing the regularization parameter $\lambda$ using leave-one-out cross validation (LOOCV) – i.e. by minimizing the leave-one-out error (LOOE). Compare the optimum $\lambda$ and test set error with the results obtained in (a). Are these different and, if they are, why?

(c) [use GURLS]: Do the same thing with an RBF kernel (dual formulation), i.e. a Gaussian of parameter $\sigma$, choosing both the regularization parameter $\lambda$ and the kernel parameter $\sigma$ to minimize LOOE. Report the train and test set error for the optimum $\sigma$ and compare with (a) and (b). (Note: You can specify the number of values GURLS will try for $\lambda$ and $\sigma$.)

(d) [Extra Credit] Plot the decision boundaries overlaid on the training set points (in two dimensions), for the linear and RBF kernel you trained (using the optimum, chosen values of $\lambda$ and $\sigma$).

Write-ups/Deliverable: Please include the following:

- MATLAB scripts for (a), (b), (c) and (optionally) (d), each part in a separate script.
- Figures/plots and errors reported in terms of percentage of mis-classified points.
References


