COMS 4771 HW2

Due: Thu Jun 11, 2015

A printed copy of the homework is due at 5:30pm in class. You must show your work to receive full credit.

1 **[Classification with 'unsure' option]** Often it is beneficial to construct classifiers which predict a label only when they are confident, and have an option to output 'unsure' if they are less confident in exchange to incurring a penalty. If the penalty for outputting 'unsure' is not too high, it may be a desirable action. Consider the penalty function

$$\varrho(\hat{y}; Y = y) = \begin{cases} 0 & \hat{y} = y \\ \lambda_u & \hat{y} = \text{'unsure'} \\ \lambda_s & \text{otherwise} \end{cases}$$

where λ_u is the penalty incurred for outputting 'unsure' and λ_s is the penalty incurred for mispredicting. Consider a joint distribution over the data X and labels Y.

(i) For a given test example x, what is the minimum possible penalty a classifier can yield? (Hint: there are multiple cases depending on the value of $P(Y|X = x)$.)

(ii) What happens if $\lambda_u = 0$? What happens if $\lambda_u \geq \lambda_s$?

2 **[Constrained optimization and duality]**

(i) Show that the distance from the hyperplane $g(x) = w \cdot x + w_0 = 0$ to a point x_a is $|g(x_a)|/\|w\|$ by minimizing the squared distance $\|x - x_a\|^2$ subject to the constraint $g(x) = 0$.

Consider the optimization problem

$$\min_{x \in \mathbb{R}^d} \|x\|$$
$$\text{such that} \sum_{i=1}^{d} x_i \geq 5$$

(ii) Is this optimization problem a convex optimization problem? why or why not?

(iii) What is the Lagrange dual of this problem?

(iv) Does strong duality hold? why or why not?

3 **[Making data linearly separable by feature space mapping]** Consider the infinite dimensional feature space mapping

$$\begin{aligned}\Phi_\sigma : \mathbb{R} &\to \mathbb{R}^\infty \\ x &\mapsto \Big(\max\Big\{0, 1 - |\frac{\alpha - x}{\sigma}|\Big\}\Big)_{\alpha \in \mathbf{R}}.\end{aligned}$$

(It may be helpful to sketch the function $f(\alpha) := \max\{0, 1 - |\alpha|\}$ for understanding the mapping and answering the questions below)

(i) Show that for any n distinct points $x_1, \ldots, x_n$, there exists a $\sigma > 0$ such that the mapping Φ_σ can linearly separate *any* binary labeling of the n points.

(ii) Show that one can efficiently compute the dot products in this feature space, by giving an analytical formula for $\Phi_\sigma(x) \cdot \Phi_\sigma(x')$ for arbitrary points x and x'.

4 **[Perceptron case study]** We shall study the relative performance of different variations of the Perceptron algorithm on the handwritten digits dataset from HW1.

Consider a sequence of training data $(x_1, y_1), \ldots, (x_n, y_n)$ in an arbitrary but fixed order (the labels y_i are assumed to be binary in $\{-1, +1\}$).

Perceptron V0
learning:
- Initialize $w_0 := 0$
- for $t = 1, \ldots, T$
- pick example (x_i, y_i), where $i = (t \bmod n + 1)$
- if $y_i(w_{t-1} \cdot x_i) < 0$
- $w_t := w_{t-1} + y_i x_i$
- else
- $w_t := w_{t-1}$

classification:
$f(x) :=$ sign$(w_T \cdot x)$

Perceptron V1
learning:
- Initialize $w_0 := 0$
- for $t = 1, \ldots, T$
- pick example (x_i, y_i), such that $i := \arg\min_j \big(y_j w_{t-1} \cdot x_j\big)$
- if $y_i(w_{t-1} \cdot x_i) < 0$
- $w_t := w_{t-1} + y_i x_i$
- else
- $w_T := w_{t-1}$; terminate

classification:
$f(x) :=$ sign$(w_T \cdot x)$

Perceptron V2

learning:

- Initialize $w_1 := 0$, $c_0 := 0$, k := 1
- for $t = 1, \ldots, T$
-   pick example (x_i, y_i), where $i = (t \bmod n + 1)$
-   if $y_i(w_k \cdot x_i) < 0$
-     $w_{k+1} := w_k + y_i x_i$
-     $c_{k+1} := 1$
-     $k := k + 1$
-   else
-     $c_k := c_k + 1$

classification:

$f(x) :=$ sign$\big(\sum_{i=1}^{k} c_k$ sign$(w_k \cdot x)\big)$

(i) Implement the three variations of the Perceptron algorithm for the 10-way digit classification problem.
You must submit your code to the TA to receive full credit.

(ii) Which Perceptron version is better for classification? You must justify your answer with appropriate performance graphs demonstrating the superiority of one classifier over the other. Example things to consider: you should evaluate how the classifier behaves on a holdout test sample for various splits of the data; how does the training sample size and the number of passes affects the classification performance.

(iii) Implement the Kernel Perceptron as described in lecture with a high degree (say, 5 to 10) polynomial kernel. How does it affect the classification on test data?