Statistical Machine Learning (GU4241/GR5241)
Spring 2017
https://courseworks.columbia.edu

Cynthia Rush
cgr2130
Peter Lee, Gabriel Loaiza
jl4304, gl2480

MIDTERM EXAM

Total time: 75 minutes. To be taken in-class, Tuesday 7 March 2017.

Do not open this exam until instructed. Carefully read the following instructions. You may not receive help from your neighbors, your friends, the internet, or any other source beyond your own knowledge of the material and your reference sheet. If you do so, you will receive a 0 grade on the midterm and will possibly fail the class or face expulsion from the program.

Write your name, UNI, and the course title on the cover of the blue book. All solutions should be written in the accompanying blue book. No other paper (including this exam sheet) will be graded. **To receive credit for this exam, you must submit blue book with the exam paper placed inside.** As reference you may use one sheet of 8.5×11 in paper, on which any notes can be written (front and back). No other materials are allowed (including calculators, textbooks, computers, and other electronics). To receive full credit on multi-point problems, you must explain how you arrived at your solutions. Each problem is divided up into several parts. Many parts can be answered independently, so if you are stuck on a particular part, you may wish to skip that part and return to it later. Good luck.

1. (25 points) Please briefly explain your answer for the following questions.

   (a) (3 points) Consider the convex optimization problem over $x \in \mathbb{R}^2$:

   $$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) = 0 \end{aligned}$$

   Consider a point x^* with $\nabla f(x^*) = \begin{bmatrix} 1.2 \\ 0.7 \end{bmatrix}$ and $\nabla g(x^*) = \begin{bmatrix} 3.6 \\ 2.1 \end{bmatrix}$. Is x^* a minimum?

   (b) (3 points) Consider the convex optimization problem over $x \in \mathbb{R}^2$:

   $$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g(x) \leq 0 \end{aligned}$$

   Consider a point x^* with $\nabla f(x^*) = \begin{bmatrix} 1.2 \\ 0.7 \end{bmatrix}$ and $\nabla g(x^*) = \begin{bmatrix} 3.6 \\ 2.1 \end{bmatrix}$. Is x^* a minimum?

   (c) (4 points) Briefly (1 sentence) describe how an ensemble method works.

   (d) (3 points) Considering a binary classifier: $y_i \in \{-1, +1\}$, if we are exclusively interested in minimizing misclassification rate, what loss function should we use?

   (e) (4 points) In the cascade classifier used to train face detection we modified the standard loss that you identified above. In what we did we modify the loss and why did we do this?

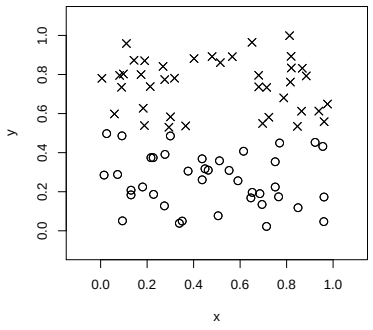
   (f) (5 points) Assume we know the class conditional distributions of the data: $p(\mathbf{x}|y = +1)$ is exactly spherical Gaussians with mean vector μ_+ and covariance $\sigma^2 I$, and similar for the -1 class: $p(\mathbf{x}|y = -1) = \mathcal{N}\left(x; \mu_-, \sigma^2 I\right)$. Notice that σ is the same for both classes, but the mean vectors are not. Describe what happens to the misclassification rate for different values of μ_+ , μ_- , σ^2. More specifically under what scenario would a classifier achieve a low misclassification rate, and conversely a high misclassification rate? Justify your answer.

   (g) (3 points) If the data is distributed according to the above assumptions, what method should we use to learn the Bayes-optimal classifier?

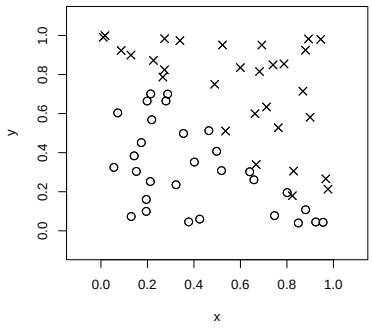
2. (16 points) Decision trees.

(a) (8 points) For each of the following data sets, explain whether or not a basic decision tree of depth 2 will excel in classifying the data. If not, propose a classifier that will.

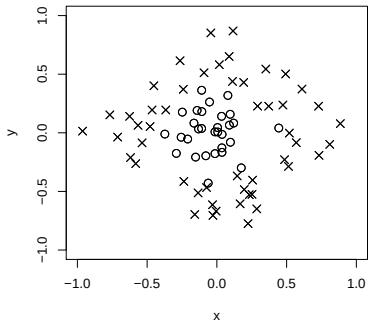
[A]



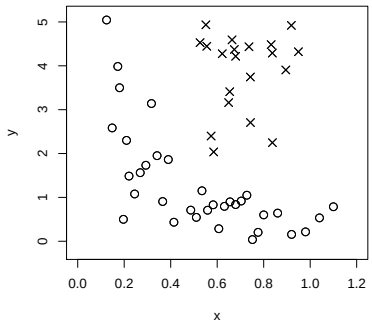
[B]



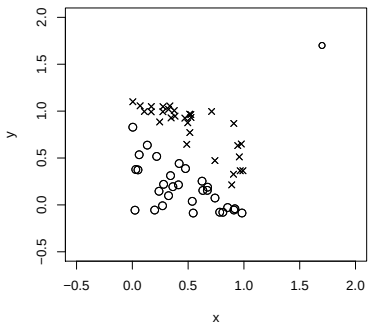
[C]



[D]



(b) (4 points) Consider training AdaBoost on the following data set, where the data are linearly separable except for the existence of an outlier. What would happen to the weight assigned to that outlier after many boosting iterations? Would you conclude that AdaBoost is robust to outliers?



(c) (4 points) In AdaBoost, would you stop the iteration if the error rate of the current weak learner on the weighted training data is 0? Explain.

3. (20 points) Gambling.

A particular gambling scheme involves n rounds of play. In the first round, the payoff is $X_1 \sim \exp(\theta, \lambda)$ for some payoff parameters θ and λ. For the ith round, the payoff is $X_i | X_{i-1} \sim \exp(\theta + X_{i-1}, \lambda + X_{i-1})$. Recall: we say $X \sim \exp(a, b)$ if:

$$f_X(x) = ae^{-a(x-b)}\mathbb{I}\{x \geq b\}.$$

(a) (8 points) I give you the observed payouts from all n rounds, $X_1, X_2, \ldots, X_n$. What is the maximum likelihood estimate of λ, namely λ_{ML}?

(b) (6 points) Assume we know λ_{ML}. I give you the observed payouts from all n rounds $X_1, X_2, \ldots, X_n$. Write down an optimization problem for θ_{ML}. Be sure to transform the optimization to a more computationally tractable form.

(c) (6 points) Write down an optimization algorithm that will optimize the function from the previous part. There are several choices; you may choose the simplest. Describe how the algorithm proceeds, and write an expression for the update rule, which should only involve θ, λ_{ML}, and the data $X_1, X_2, \ldots, X_n$.

4. (14 points) Naive Bayes

(a) (6 points) Consider the following data set with covariates $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$: $x_1 \in \{True, False\}$, $x_2 \in \{Red, Green, Blue\}$, and class labels, $y \in \{-1, +1\}$:

y	x_1	x_2
-1	False	Green
-1	True	Green
-1	False	Blue
-1	True	Red
$+1$	False	Green
$+1$	False	Green
$+1$	False	Green
$+1$	True	Blue

Train the Naive Bayes classifier on this data: estimate all necessary probability distributions.

(b) (6 points) Classify the training data.

(c) (2 points) Calculate the misclassification rate of this classifier on the training data.

5. (25 points) Kernel perceptron.

The perceptron has parameter $z \in \mathbb{R}^{d+1}$, and makes predictions of +1 or -1 for the input x using the classification function:

$$f(x) = \text{sgn}\left(\left\langle \begin{bmatrix}1\\x\end{bmatrix}, z \right\rangle\right).$$

To learn from a labeled dataset of (x_i, y_i), $i = 1, \ldots, n$ where $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$ (with learning rate $\alpha = 1$), the batch perceptron (i.e., all data points in one batch) learns by repeatedly updating z using the training rule:

$$z^{k+1} = z^k - \sum_{i=1}^{n} \mathbb{I}\{f(x_i) \neq y_i\} \cdot (-y_i) \begin{bmatrix}1\\x_i\end{bmatrix}.$$

Recall that the above update is the gradient descent update rule for the perceptron objective:

$$C(z) = \sum_{i=1}^{n} \mathbb{I}\{f(x_i) \neq y_i\} \cdot \left|\left\langle z, \begin{bmatrix}1\\x_i\end{bmatrix}\right\rangle\right|.$$

(a) (5 points) The perceptron can also be trained as a single-sample algorithm, updating z one training data point at a time. Write the training rule for single-sample perceptron, i.e., how to compute z^{k+1} given z^k.

(b) (5 points) Using the single sample perceptron update rule and beginning the updates at $z^0 = 0$, after some number of iterations, z can be written as $z = \sum_{i=1}^{n} a_i y_i \begin{bmatrix}1\\x_i\end{bmatrix}$. What is a_i?

(c) (10 points) We saw that the kernel trick produces non-linear SVM with a relatively small change to the linear SVM. We want to kernelize the perceptron algorithm to provide non-linear decision boundaries. With a mapping ϕ to some feature space $\mathcal{F}$, we rewrite the classifier as:

$$f(x) \;=\; \text{sgn}\left(\left\langle \phi(z), \phi\left(\begin{bmatrix}1\\x\end{bmatrix}\right)\right\rangle_{\mathcal{F}}\right) \;=\; \text{sgn}\left(\sum_{i=1}^{n} w_i k(x_i, x)\right).$$

Implicit in the second equality is that we have enforced something called the "representer theorem", namely $\phi(z) = \sum_{i=1}^{n} w_i \phi\left(\begin{bmatrix}1\\x_i\end{bmatrix}\right)$. Thus we now have parameters $w_1, ..., w_n$; which, for convenience, we can initialize to $w_i^0 = 0$ for $i = 1, \ldots, n$.

What is the training rule for kernel perceptron; that is, how do we update w^{k+1} from w^k?

(d) (5 points) Do you expect kernel perceptron would generalize well? Why or why not?