10-701 Midterm Exam, Fall 2007

1. Personal info:
   - Name:
   - Andrew account:
   - E-mail address:
2. There should be 17 numbered pages in this exam (including this cover sheet).
3. You can use any material you brought: any book, class notes, your print outs of class materials that are on the class website, including my annotated slides and relevant readings, and Andrew Moore's tutorials. You cannot use materials brought by other students. Calculators are not necessary. Laptops, PDAs, phones and Internet access are not allowed.
4. If you need more room to work out your answer to a question, use the back of the page and clearly mark on the front of the page if we are to look at what's on the back.
5. Work efficiently. Some questions are easier, some more difficult. Be sure to give yourself time to answer all of the easy ones, and avoid getting bogged down in the more difficult ones before you have answered the easier ones.
6. Note there are extra-credit sub-questions. The grade curve will be made without considering students' extra credit points. The extra credit will then be used to try to bump your grade up without affecting anyone else's grade.
7. You have 80 minutes.
8. Good luck!

Question	Topic	Max. score	Score
1	Short questions	20 + 0.1010 extra	
2	Loss Functions	12	
3	Kernel Regression	12	
4	Model Selection	14	
5	Support Vector Machine	12	
6	Decision Trees and Ensemble Methods	30	

1 [20 Points] Short Questions

The following short questions should be answered with at most two sentences, and/or a picture. For yes/no questions, make sure to provide a *short* justification.

1. [2 point] Does a 2-class Gaussian Naive Bayes classifier with parameters μ_{1k}, σ_{1k}, μ_{2k}, σ_{2k} for attributes $k = 1, ..., m$ have exactly the same representational power as logistic regression (i.e., a linear decision boundary), given no assumptions about the variance values σ_{ik}^2?

2. [2 points] For linearly separable data, can a small slack penalty ("C") hurt the training accuracy when using a linear SVM (no kernel)? If so, explain how. If not, why not?

3. [3 points] Consider running AdaBoost with Multinomial Naive Bayes as the weak learner for two classes and k binary features. After t iterations, of AdaBoost, how many parameters do you need to remember? In other words, how many numbers do you need to keep around to predict the label of a new example? Assume that the weak-learner training error is non-zero at iteration t. Don't forget to mention where the parameters come from.

4. [2 points] In boosting, would you stop the iteration if the following happens? Justify your answer with at most two sentences each question.

   - The error rate of the combined classifier on the original training data is 0.

- The error rate of the current weak classifier on the weighted training data is 0.

5. [4 points] Given n linearly independent feature vectors in n dimensions, show that for any assignment to the binary labels you can always construct a linear classifier with weight vector w which separates the points. Assume that the classifier has the form $sign(w \cdot x)$. Note that a square matrix composed of linearly independent rows is invertible.

6. [3 points] Construct a one dimensional classification dataset for which the Leave-one-out cross validation error of the One Nearest Neighbors algorithm is always 1. Stated another way, the One Nearest Neighbor algorithm never correctly predicts the held out point.

7. [2 points] Would we expect that running AdaBoost using the ID3 decision tree learning algorithm (without pruning) as the weak learning algorithm would have a better true error rate than running ID3 alone (i.e., without boosting (also without pruning))? Explain.

8. [1 point] Suppose there is a coin with unknown bias p. Does there exist some value of p for which we would expect the maximum a-posteriori estimate of p, using a $Beta(4, 2)$ prior, to require more coin flips before it is close to the true value of p, compared to the number of flips required of the maximum likelihood estimate of p? Explain.
(The $Beta(4, 2)$ distribution is given in the figure below.)

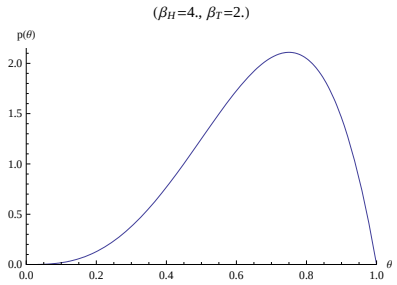


Figure 1: $Beta(4, 2)$ distribution

9. [1 point] Suppose there is a coin with unknown bias p. Does there exist some value of p for which we would expect the maximum a-posteriori estimate of p, using a $Uniform([0, 1])$ prior, to require more coin flips before it is close to the true value of p, compared to the number of flips required of the maximum likelihood estimate of p? Explain.

10. [0.1010 extra credit] Can a linear classifier separate the positive from the negative examples in the dataset below? Justify.

Colbert
for
president

U2
Loosing my religion

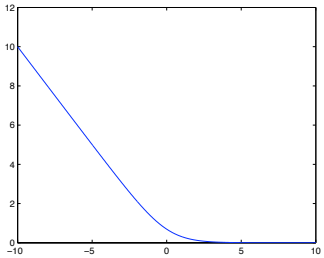
The Beatles
There is a season...
Turn! Turn! Turn!

Nirvana
Grunge

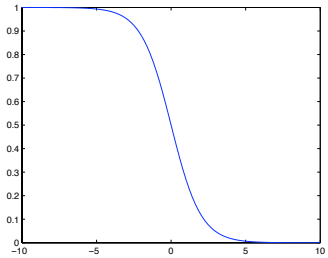
2 [12 points] Loss Function

Generally speaking, a classifier can be written as $H(x) = \text{sign}(F(x))$, where $H(x) : \mathbb{R}^d \to \{-1, 1\}$ and $F(x) : \mathbb{R}^d \to \mathbb{R}$. To obtain the parameters in $F(x)$, we need to minimize the loss function averaged over the training set: $\sum_i L(y^i F(x^i))$. Here L is a function of $yF(x)$. For example, for linear classifiers, $F(x) = w_0 + \sum_{j=1}^{d} w_j x_j$, and $yF(x) = y(w_0 + \sum_{j=1}^{d} w_j x_j)$

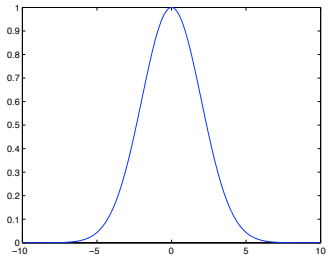
1. [4 points] Which loss functions below are appropriate to use in classification? For the ones that are not appropriate, explain why not. In general, what conditions does L have to satisfy in order to be an appropriate loss function? The x axis is $yF(x)$, and the y axis is $L(yF(x))$.



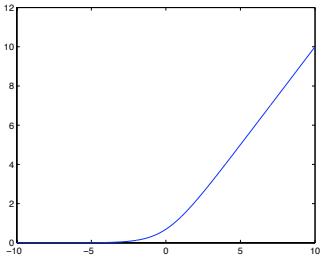
(a)



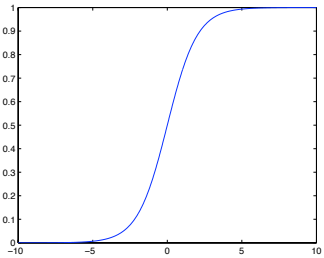
(b)



(c)



(d)



(e)

2. [4 points] Of the above loss functions appropriate to use in classification, which one is the most robust to outliers? Justify your answer.

3. [4 points] Let $F(x) = w_0 + \sum_{j=1}^{d} w_j x_j$ and $L(yF(x)) = \frac{1}{1+\exp(yF(x))}$. Suppose you use gradient descent to obtain the optimal parameters w_0 and w_j. Give the update rules for these parameters.

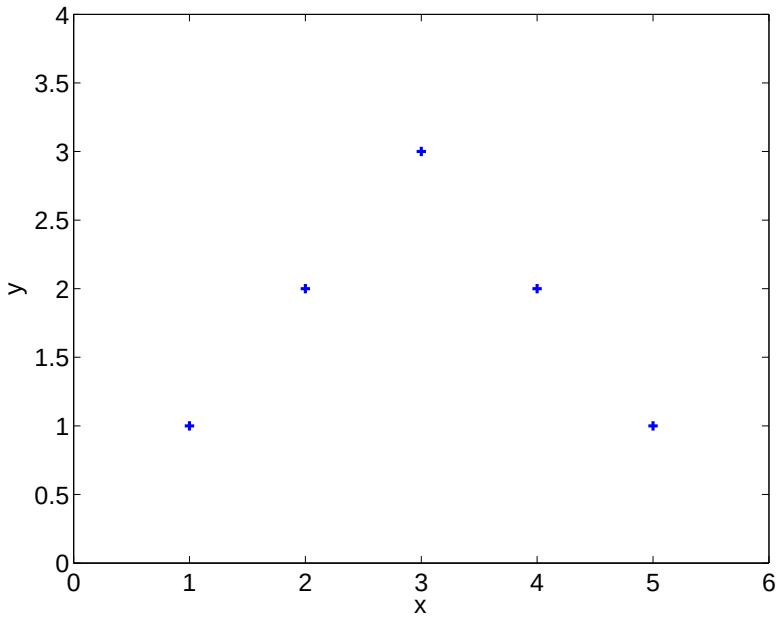
3 [12 points] Kernel Regression, k-NN

1. [4 points] Sketch the fit Y given X for the dataset given below using kernel regression with a box kernel

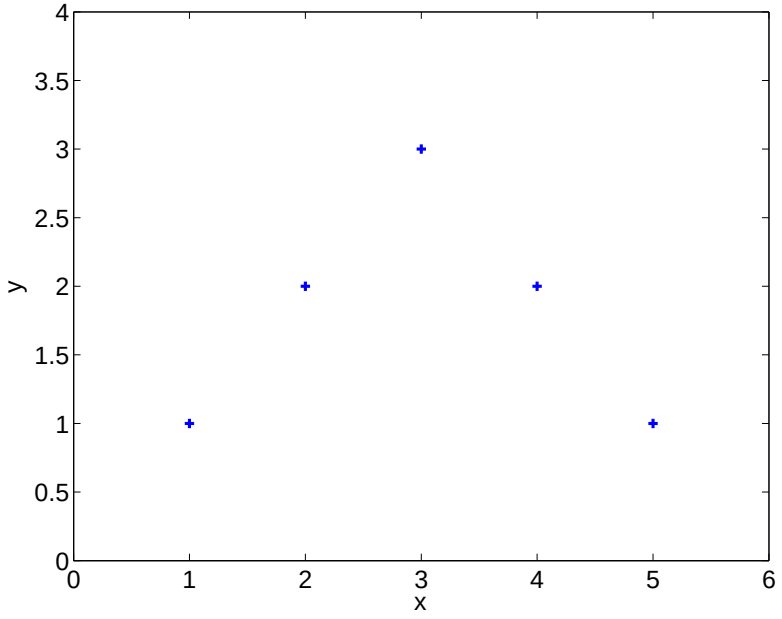
$$K(x_i, x_j) = I(-h \leq x_i - x_j < h) = \begin{cases} 1 & if -h \leq x_i - x_j < h \\ 0 & otherwise \end{cases}$$

for $h = 0.5, 2$.

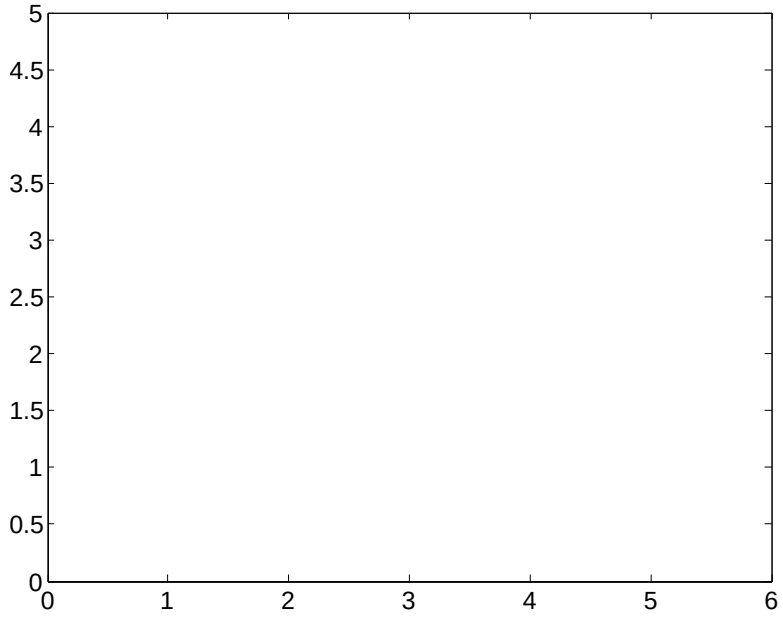
- $h = 0.5$



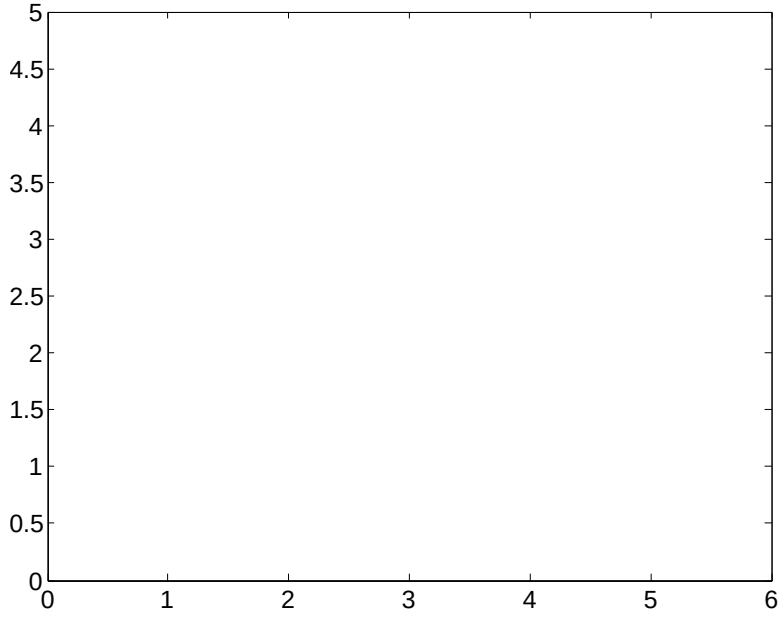
- $h = 2$



2. [4 points] Sketch or describe a dataset where kernel regression with the box kernel above with $h = 0.5$ gives the same regression values as 1-NN but not as 2-NN in the domain $x \in [0, 6]$ below.



3. [4 points] Sketch or describe a dataset where kernel regression with the box kernel above with $h = 0.5$ gives the same regression values as 2-NN but not as 1-NN in the domain $x \in (0, 6)$ below.



4 [14 Points] Model Selection

A central theme in machine learning is model selection. In this problem you will have the opportunity to demonstrate your understanding of various model selection techniques and their consequences. To make things more concrete we will consider the dataset $\mathcal{D}$ given in **??** consisting of n independent identically distributed observations. The features of $\mathcal{D}$ consist of pairs $(x_1^i, x_2^i) \in \mathbb{R}^2$ and the observations $y^i \in \mathbb{R}$ are continuous valued.

$$\mathcal{D} = \{((x_1^1, x_2^1), y^1), ((x_1^2, x_2^2), y^2), \ldots, ((x_1^n, x_2^n), y^n)\} \tag{1}$$

Consider the abstract model given **??**. The function f_{θ_1,θ_2} is a mapping from the features in $\mathbb{R}^2$ to an observation in $\mathbb{R}^1$ which depends on two parameters θ_1 and θ_2. The ϵ^i correspond to the noise. Here we will assume that the $\epsilon^i \sim N(0, \sigma^2)$ are independent Gaussians with zero mean and variance σ^2.

$$y^i = f_{\theta_1,\theta_2}(x_1^i, x_2^i) + \epsilon^i \tag{2}$$

1. [4 Points] Show that the log likelihood of the data given the parameters is equal to **??**.

$$l(D; \theta_1, \theta_2) = -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y^i - f_{\theta_1,\theta_2}(x_1^i, x_2^i))^2 - n \log\left(\sqrt{2\pi}\sigma\right) \tag{3}$$

   Recall the probability density function of the $N(\mu, \sigma^2)$ Gaussian distribution is given by **??**.

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{4}$$

2. [1 Point] If we disregard the parts that do not depend on f_{θ_1,θ_2} and Y the negative of the log-likelihood given in **??** is equivalent to what commonly used loss function?

3. [2 Points] Many common techniques used to find the maximum likelihood estimates of θ_1 and θ_2 rely on our ability to compute the gradient of the log-likelihood. Compute the gradient of the log likelihood with respect to θ_1 and θ_2. Express you answer in terms of:

$$y^i, \quad f_{\theta_1,\theta_2}(x_1^i, x_2^i), \quad \frac{\partial}{\partial\theta_1} f_{\theta_1,\theta_2}(x_1^i, x_2^i), \quad \frac{\partial}{\partial\theta_2} f_{\theta_1,\theta_2}(x_1^i, x_2^i)$$

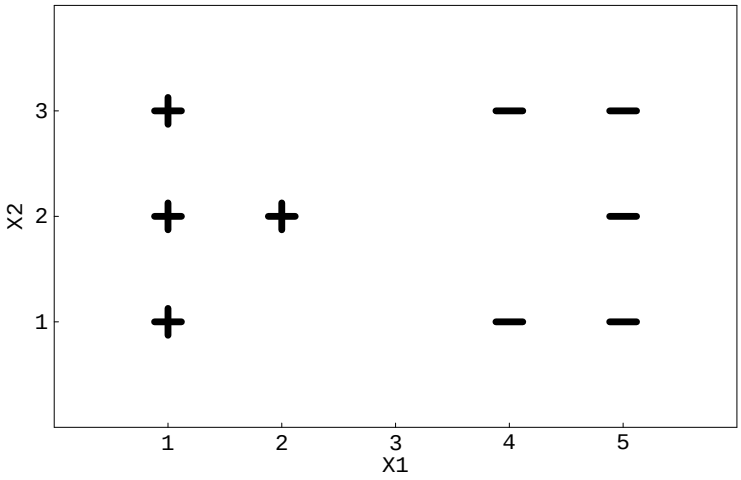
4. [2 Points] Given the learning rate η, what update rule would you use in gradient descent to *maximize* the likelihood.

5. [3 Points] Suppose you are given some function h such that $h(\theta_1, \theta_2) \in \mathbb{R}$ is large when f_{θ_1,θ_2} is complicated and small when f_{θ_1,θ_2} is simple. Use the function h along with the negative log-likelihood to write down an expression for the regularized loss with parameter λ.

6. [2 Points] For small and large values of λ describe the bias variance trade off with respect to the regularized loss provided in the previous part.

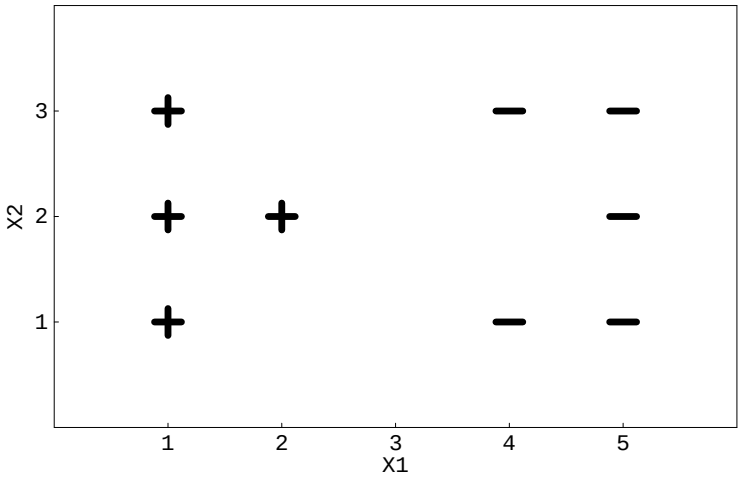
5 [12 points] Support Vector Machine

1. [2 points] Suppose we are using a linear SVM (i.e., no kernel), with some large C value, and are given the following data set.



Draw the decision boundary of linear SVM. Give a brief explanation.

2. [3 points] In the following image, circle the points such that removing that example from the training set and retraining SVM, we would get a different decision boundary than training on the full sample.

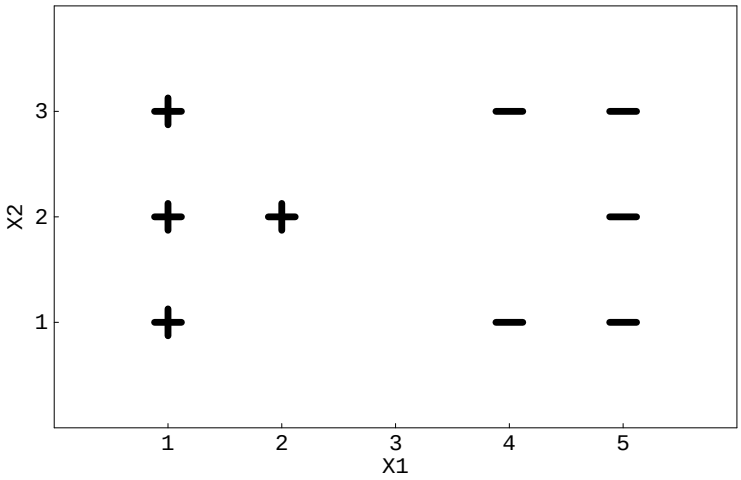


You do not need to provide a formal proof, but give a one or two sentence explanation.

3. [3 points] Suppose instead of SVM, we use regularized logistic regression to learn the classifier. That is,

$$(w, b) = \arg \min_{w \in \mathbb{R}^2, b \in \mathbb{R}} \frac{\|w\|^2}{2} - \sum_i \mathbb{1}[y^{(i)} = 0] \ln \frac{1}{1 + e^{(w \cdot x^{(i)} + b)}} + \mathbb{1}[y^{(i)} = 1] \ln \frac{e^{(w \cdot x^{(i)} + b)}}{1 + e^{(w \cdot x^{(i)} + b)}}.$$

In the following image, circle the points such that removing that example from the training set and running regularized logistic regression, we would get a different decision boundary than training with regularized logistic regression on the full sample.



You do not need to provide a formal proof, but give a one or two sentence explanation.

4. [4 points] Suppose we have a kernel $K(\cdot, \cdot)$, such that there is an implicit high-dimensional feature map $\phi : \mathbb{R}^d \to \mathbb{R}^D$ that satisfies $\forall x, z \in \mathbb{R}^d,\ K(x, z) = \phi(x) \cdot \phi(z)$, where $\phi(x) \cdot \phi(z) = \sum_{i=1}^{D} \phi(x)_i \phi(z)_i$ is the dot product in the D-dimensional space.

   Show how to calculate the Euclidean distance in the D-dimensional space

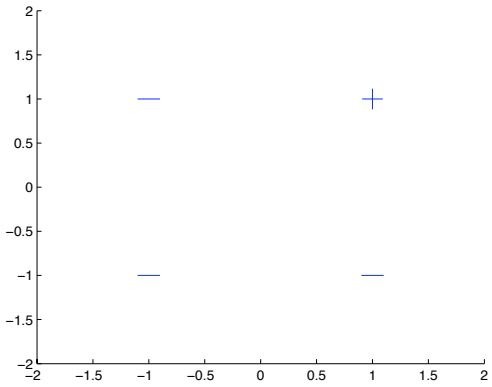
$$\|\phi(x) - \phi(z)\| = \sqrt{\sum_{i=1}^{D} (\phi(x)_i - \phi(z)_i)^2}$$

   without explicitly calculating the values in the D-dimensional vectors. For this question, you should provide a formal proof.

6 [30 points] Decision Tree and Ensemble Methods

An ensemble classifier $H_T(x)$ is a collection of T weak classifiers $h_t(x)$, each with some weight α_t, $t = 1, \ldots, T$. Given a data point $x \in \mathbb{R}^d$, $H_T(x)$ predicts its label based on the weighted majority vote of the ensemble. In the binary case where the class label is either 1 or -1, $H_T(x) = \text{sgn}(\sum_{t=1}^{T} \alpha_t h_t(x))$, where $h_t(x) : \mathbb{R}^d \to \{-1, 1\}$, and $\text{sgn}(z) = 1$ if $z > 0$ and $\text{sgn}(z) = -1$ if $z \leq 0$. Boosting is an example of ensemble classifiers where the weights are calculated based on the training error of the weak classifier on the weighted training set.

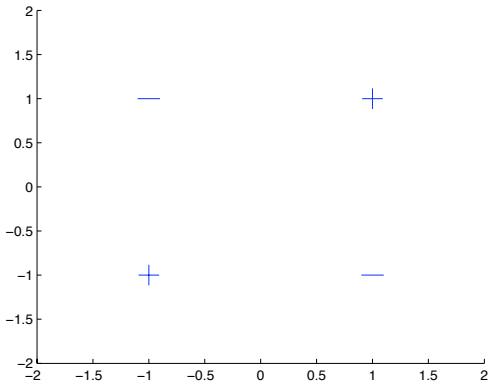
1. [10 points] For the following data set,



   - Describe a binary decision tree with the minimum depth and consistent with the data;

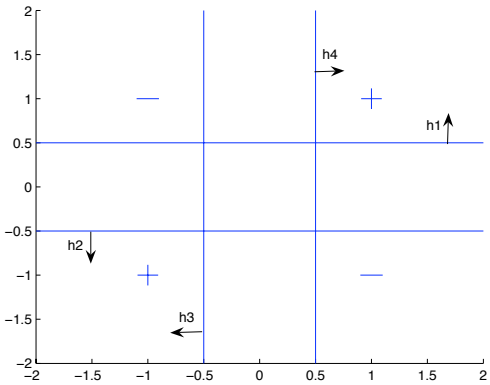
   - Describe an ensemble classifier $H_2(x)$ with 2 weak classifiers that is consistent with the data. The weak classifiers should be simple decision stumps. Specify the weak classifiers and their weights.

2. [10 points] For the following XOR data set,



- Describe a binary decision tree with the minimum depth and consistent with the data;

- Let the ensemble classifier consist of the four binary classifiers shown below (the arrow means that the corresponding classifier classifies every data point in that direction as +), prove that there are no weights $\alpha_1, \ldots, \alpha_4$, that make the ensemble classifier consistent with the data.



3. [10 points] Suppose that for each data point, the feature vector $x \in \{0,1\}^m$, i.e., x consists of m binary valued features, the class label $y \in \{-1,1\}$, and the true classifier is a majority vote over the features, i.e. $y = \text{sgn}(\sum_{i=1}^{m}(2x_i - 1))$, where x_i is the i^{th} component of the feature vector.

   - Describe a binary decision tree with the minimum depth and consistent with the data. How many leaves does it have?

   - Describe an ensemble classifier with the minimum number of weak classifiers. Specify the weak classifiers and their weights.