**ECE/CS 498 DSU/DSG Spring 2020**

**In-Class Activity 6**

NetID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The purpose of the in-class activity is for you to:

1. Review concepts related to SVM and neural networks
2. Work out steps in backpropagation for optimization of a neural network

# Support Vector Machine

## Linear decision boundary

### Sketch the hyperplane . Indicate the set of points for which , as well as the set of points for which .

## Non-linear decision boundary

### Sketch the curve .

### On your sketch, indicate the region for which , as well as the region for which .

### Suppose that a classifier assigns an observation to the blue class if , and to the red class otherwise. To what class is the observation classified? ?

### The decision boundary equation in c) is not linear in terms of an input (, ) since it has and terms. However, suppose that we instead consider (, , , ). This might be the case after applying a kernel transformation to the dataset. Is the decision boundary equation linear then? What might be a reason for applying this kernel transformation?

## Hard margin SVM

Suppose we are learning a hard margin SVM with two real-valued features and binary label (represented by △ and ○, respectively). The training data is pictured in the figure below. Our linear classifier takes the form .



H1

H2

### According to the maximum margin principle, identify the support vectors, and sketch the decision boundary of the trained SVM.

### Suppose hyperplane takes the form , write down the equations for .

### The constraints for linear hard margin SVM can be written as . Explain why.

### What condition do the data points have to satisfy such that a feasible exists?

### Calculate the distance between and .

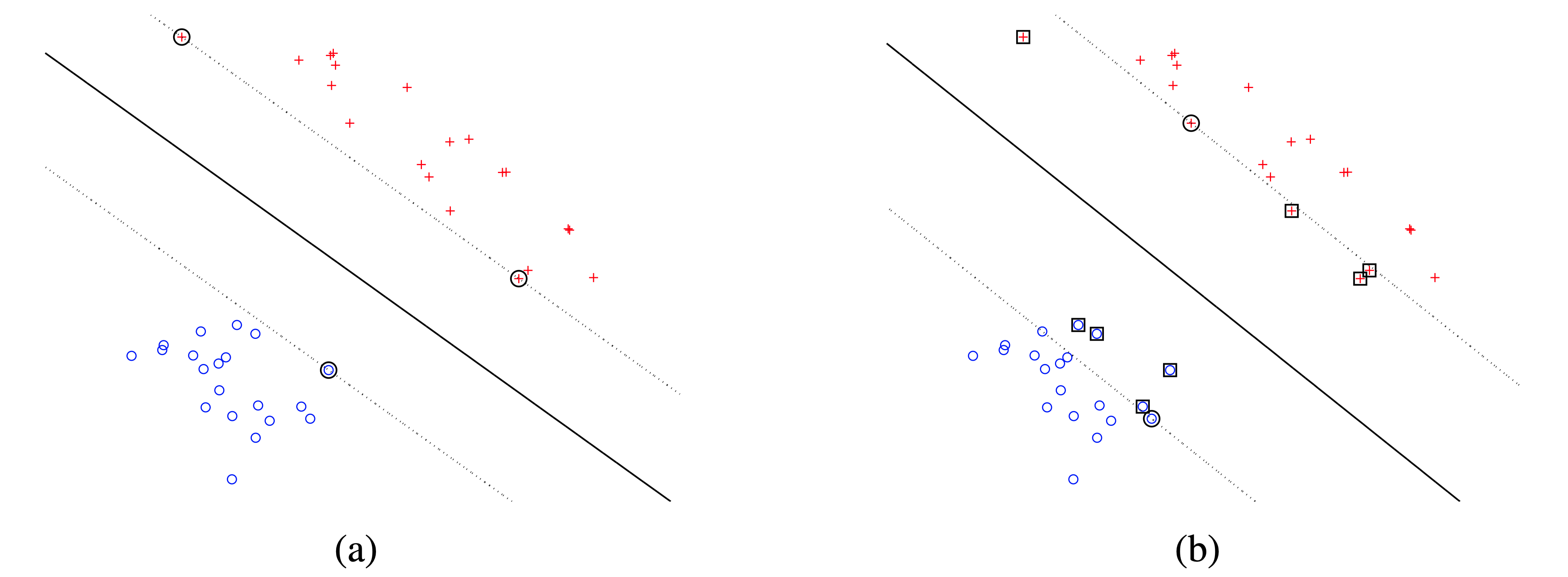
### Based on your answer to the above questions, write down the optimization problem whose solution is hard margin SVM.

### For the following data points corresponding to class , one of the following , gives the correct SVM decision boundary (). Which one is it? Show your work.

## Soft margin SVM

Recall the program for solving the soft margin SVM:

We have plotted the SVM solutions for a training dataset in Figure (a) and (b) corresponding to two values of :



### Indicate non-zero s on both plots (a) and (b).

### Which figure corresponds to a larger value of ? Explain why.

The unconstrained form of the above optimization problem is given as:

### Draw the function for scalar variable z. What is the derivative ? Draw the derivative.

### Compute the gradient of this unconstrained program w.r.t . [Hint: Think of in part (c) as , you can express max in terms of an indicator function]

### Suppose you are training your SVM using gradient descent and the gradient derived in (d), the original gradient is . The first iteration we trained on data point ; the second iteration we trained on data point , , assuming both C and learning-rate to be 1 calculate the gradient after these two iterations.

# Neural Networks

## Partial derivatives

Consider the network shown in the figure below. All the hidden units use the ReLU- . We are trying to minimize a cost function which depends only on the activation of the output unit . The unit (marked with a \*) receives an input of on a training iteration, so its output is . Based only on this information, which of the following weight derivatives are guaranteed to be for this training case? Write YES or NO for each. Justify your answers informally. (Hint: don’t work through the backprop computations. Instead think about what the partial derivatives really mean.)

# 

## Backpropagation

## The neural network considered in this question has three input neurons, one hidden layer with two neurons, and one output layer with two neurons. and are bias terms.



### Suppose , , and similar relationships hold for the other neurons in the hidden/output layer. Assume the current parameters of the networks . Given input , use forward propagation to find out the value of You can think of writing a Python program.

### Let represent the true labels. Define the sum of squared loss . **Write down the partial derivatives according to the chain rule**. Example for is given below.

For , look at the two paths which lead from to . Backpropagation includes both the paths in the calculation.

### Suppose , learning rate . Calculate the updated after one iteration of backpropagation. Use 1,

## Neural Network Playground (for you to explore)

<https://developers.google.com/machine-learning/crash-course/introduction-to-neural-networks/playground-exercises>