Sta-395 Intro to ML (Practice Exam)

Your Name: _____

Directions

- Answer each question using *no more than the specified number of sentences*, and do not attempt to avoid these guidelines using run-on sentences. Answers that are unnecessarily verbose may be penalized.
- Do not include superfluous information in your answers, you may be penalized if you make an inaccurate statement even if you go on to provide a correct answer. Your answers should be clear, concise, and include only what is needed to answer the question that was asked.

Note: This practice exam is somewhat shorter in length than the actual exam will be. It is intended to familiarize you with the style of questions you should expect.

Formulas

- **True Positive Rate (TPR)** - $\frac{\text{True Positives}}{\text{Total Positives}}$
- **Sensitivity** and **Recall** - TPR
- **Specificity** - $1 - \text{FPR}$
- **Precision** - $\frac{\text{True Positives}}{\text{Total Predicted Positives}}$
- **Balanced Accuracy** - $\frac{\text{TPR}+\text{FPR}}{2}$
- **F1 Score** - $\frac{2*\text{Precision}*\text{Recall}}{\text{Precision}+\text{Recall}}$

Section 1

Part A: In a few sentences, explain your understanding of the *bias-variance trade off*. More specifically, why does variance increase as bias decreases? How does one go about balancing these when developing a machine learning model?

Part B: Consider the application of 5-fold cross-validation to evaluate the performance of the decision tree classification algorithm with maximum depth of 4 using a training set consisting of $n = 100$ observations.

- i: How many times must a decision tree be trained during this evaluation process?
- ii: How many data-points are used to train each decision tree involved in the evaluation process?
- iii: How many times is each data-point used as part of the training set within the evaluation process?

Part C: Consider a machine learning application that seeks to classify emails as either "spam" or "not spam". On the test set, 980 of 1000 "not spam" emails were correctly classified but only 120 of 300 "spam" emails were correctly classified.

- i: Create a confusion matrix summarizing the performance of this classifier. Consider the outcome "spam" to be the positive class.
- ii: What is the *classification accuracy* of the classifier?
- iii: What is the *F1 score* of the classifier?
- iv: Which metric, classification accuracy or F1 score, provides a more useful assessment of how effective this classifier is? Briefly explain.
- v: What would need to change about the application or the nature of the observed data in order for the other metric (whichever you did not select in part iv) to become the more useful quantity? Briefly explain.

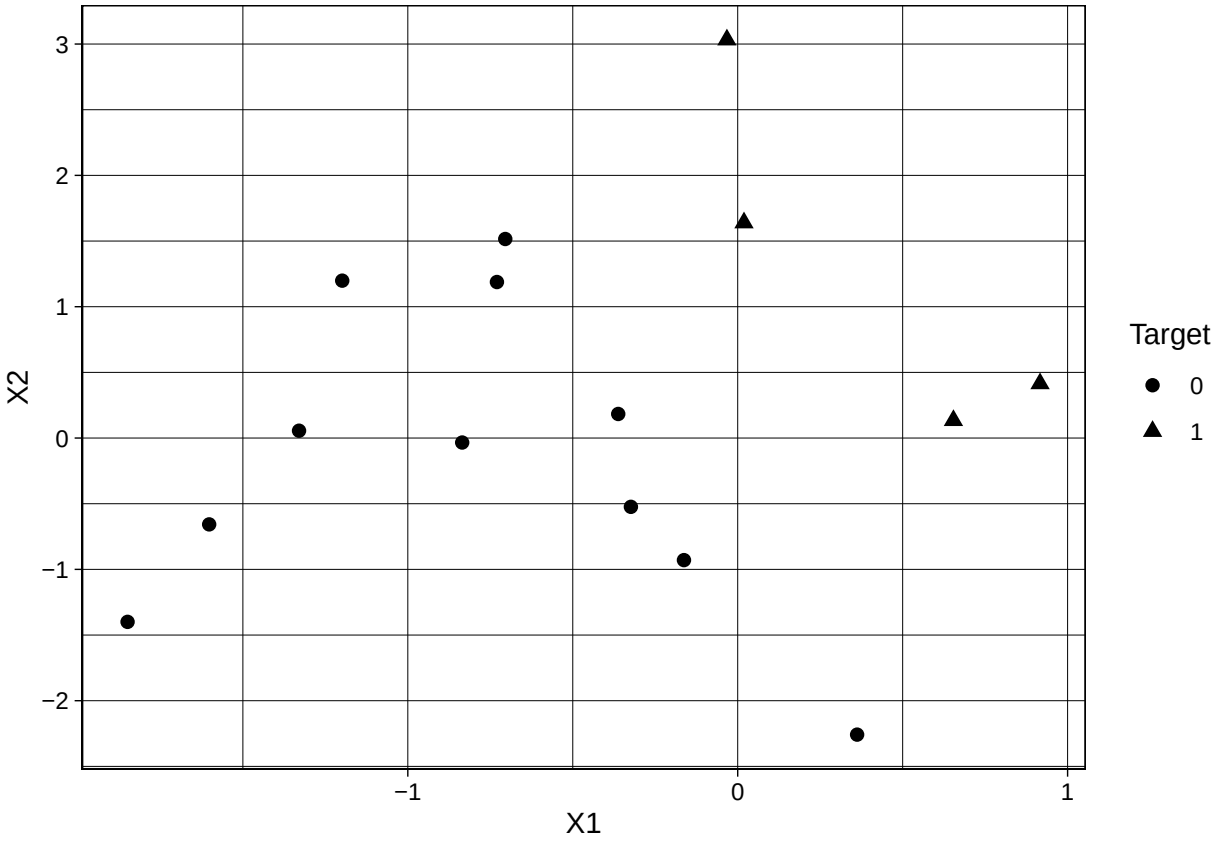
Part D: Clearly mark each of the following statements as either TRUE or FALSE. If a statement is false, briefly explain why it is incorrect.

- i: Min-max scaling can be used to modify the distributional shape of a variable before it is used in a machine learning model
- ii: Re-scaling steps should be applied to an entire data set before it is separated into training and testing sets in order to obtain the best estimate of how a modeling methodology will generalize to new data.
- iii: In k-nearest neighbors, uniform weighting allows neighbors that are closer in distance to a new data-point to have a larger influence on the algorithm's prediction than neighbors that are further away from the new data-point
- iv: In k-nearest neighbors, using more neighbors will decrease the bias of the algorithm's predictions.
- v: While the random forest algorithm benefits from bagging, or bootstrap aggregation, this is not an essential component of the algorithm.
- vi: An essential step in setting up a gradient descent algorithm is differentiating the cost function with respect to the machine learning model's unknown parameters

Section 2

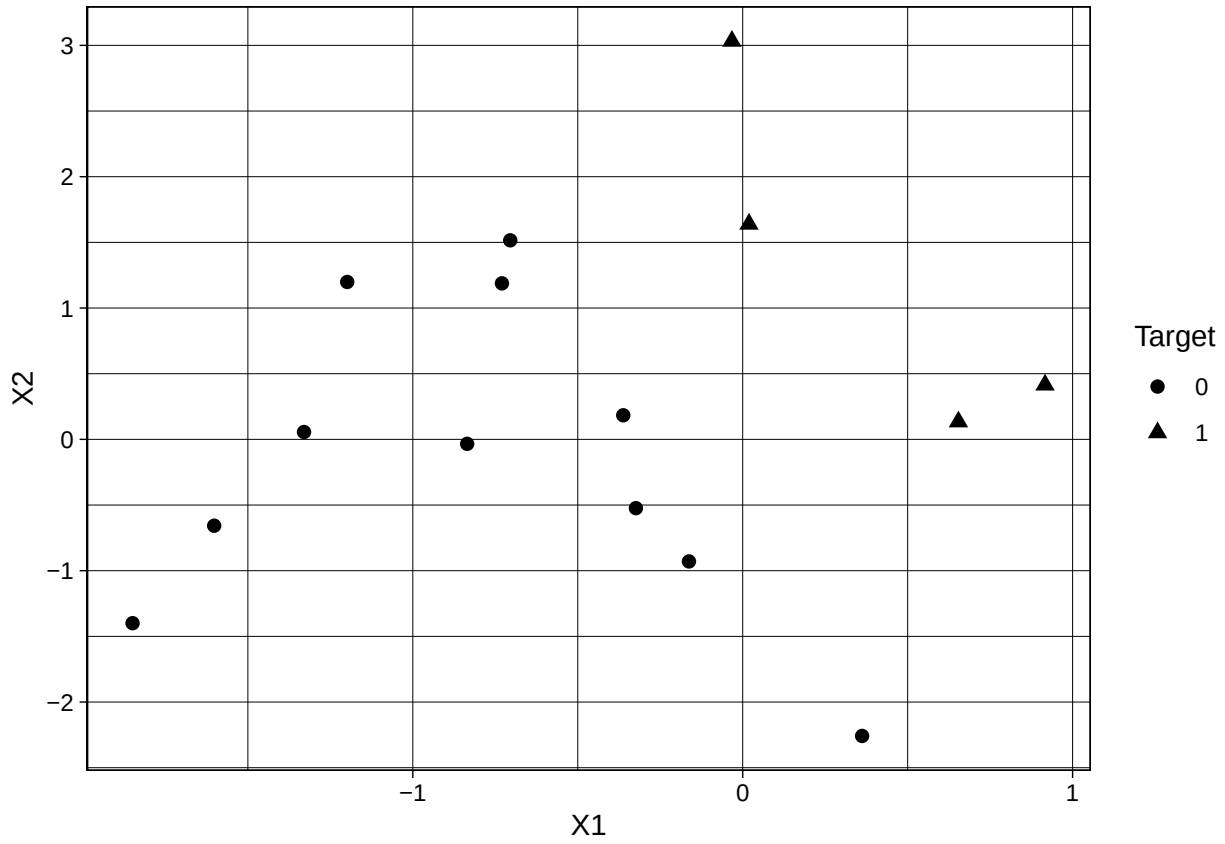
Part A: Consider the training data shown below and the goal of correctly classifying "Target" as either 1 or 0. On the given graph

- i: Sketch the decision boundary of a support vector machine classifier using a radial basis function (RBF) kernel *using a solid line*. Your boundary doesn't need to be perfect, but it should clearly illustrate that you understand the decision boundary produced by this method.
- ii: Sketch the decision boundary of a support vector machine classifier using a linear kernel *using a dashed line*. Your boundary doesn't need to be perfect, but it should clearly illustrate that you understand the decision boundary produced by this method.



Part B: Shown below are the same data used in Part A

- i: Sketch the splitting rules of a decision tree classifier with a maximum depth of 2. Your split locations don't need to be perfect, but they should clearly illustrate that you understand this method.
- ii: Consider *decreasing* the maximum depth of the classifier from Part A to a value of 1. What consequences does this change have on the *bias* and *variance* as the classification method?



Part C: Considering all of the modeling approaches from Parts A and B, which method would you recommend for a client who is interested in achieving maximum classification accuracy *on new data* that arises from the same collection procedure that gave rise to the data shown in Parts A and B? Briefly explain your answer.

Section 3

Consider the application of simple neural network consisting of one hidden layer with only one neuron and the sigmoid activation function. The input data set will involve 2 predictors. Furthermore, you may assume the outcome variable is numeric and the network is trained using the squared error cost function: Cost $= \frac{1}{n}(\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}})$

Part A: Create a sketch of this network's architecture. How many weight parameters are present in this network?

Part B: Assume that biases are used in both the hidden layer and the final layer. How many bias parameters are present in this network?

Part C: Under the simplifying assumption that we'll be training this network using batches of size $n = 1$ (ie: a single 2-dimensional vector $\mathbf{x}_i$ containing the values of each predictor for the i^{th} sample), calculate the gradient components corresponding to each weight and bias parameter involved in this network using chain rule. Clearly label each component. You may denote the sigmoid function as $\sigma()$ and its derivative as $\sigma() * (1 - \sigma())$ when forming your answer.

Part D: Briefly explain how the gradient calculated in Part C is used to update the weight parameters of the network during training.

Part E: If the gradient component for a particular weight parameter evaluates to a negative value at the current iteration of gradient descent, how would you expect the cost function to change if the value of that weight parameter is *decreased*?