Assignment 2

Applied Machine Learning

1. [20 pts] At a high-level, without entering into mathematical details, compare and contrast

the following classifiers:

• Perceptron (textbook's version) A perceptron is an algorithm that automatically learns optimal weight coefficients that is multiplied with input features in order to make the decision of whether a neuron fires (transmits a signal) or no. Perceptron can use numerical or nominal data. Perceptron algorithm does not solve an optimization problem, but is an iterative process that solves a feasibility problem. Naturally binary classifier.

• SVM – Support vector machine can be considered an extension of the perceptron algorithm where the optimization objective is to maximize the margin opposed to minimizing misclassification errors with the perceptron algorithm. The margin is defined as the distance between the separating hyperplane (decision boundary) and the training examples that are closest to this hyperplane, which are the so-called support vectors. I believe SVM can solve an optimization problem with an optimization algorithm. SVM is also binary classifier, but with multiple examples.

• Decision Tree - decision tree classifiers are attractive models if we care about interpretability. As the name "decision tree" suggests, we can think of this model as breaking down our data by making a decision based on asking a series of questions. Decision tree usually uses nominal data, and text can be used. Optimization is greedy.

• Random Forest (you have to research a bit about this classifier) – Takes the predictions of many decision trees and takes the average from those trees to increase the accuracy in regression and classification problems. Random forest usually uses nominal data.

• Which one will be the first that you would try on your dataset?

I would try using a decision tree first because it can be used for classification and regression for categorical data better than SVM and is less complicated than random forest.

2. [20 pts] Using real datasets (can also be hypothetically constructed by yourself) define thefollowing feature types, and give example values from your dataset. How would you represent these features in a computer program? (e.g., 32-bit integer? Floating point?

String?)

• Numerical – floating point float 64.

• Nominal – should be a 32-bit integer. If data is not ordinal then you’d have to use one-hot encoding to make non-ordinal nominal values into ordinal. A good example for making nominal values ordinal with one-hot encoding would be colors.

• Date – depending on what problem we’re trying to solve this would be a 32-bit integer.

• Text - String

• Image - 2d array of integers.

• Dependent variable – could be anything. String, floating point, integer…

3. [20 pts] Using online resources, research and find other classifier performance metrics which are also as common as the accuracy metric. In your own words write down the mathematical equations and the meaning of the metrics that you found.

Recall or true positive rate (TPR) is a classifier performance metric used in object classification that mathematically describes whether a specific condition exists or is absent. If a condition is present then the model is positive and if the condition is absent the model is negative.

To calculate TPR you need; true positive (TP), false negative (NF), condition positive (P), and false negative rate (FNR).

TPR = TP/P = TP/(TP +FN) = 1 – FNR.

Recall is part of the accuracy equation, but it is not a complete metric like accuracy.

4. [40 pts] Implement a correlation program from scratch to look at the correlations between

Observe that the diagonal of this matrix should have all 1's and explain why? Since the last

column can be used as the target (dependent) variable, what do you think about the

correlations between all the variables? Which variable should be the most important for

prediction of 'Chance of Admit'?