**CS6735 Final Exam (1 hour)**

**Answer *Yes* of *No* for the following questions, and explain your answer in two sentences (3 points each):**

1. **Decision tree learning algorithm ID3 adopts restriction bias.**

NO- this algorithm uses preference bias instead of restriction bias because it prefers to start doing partition from features that have highest gain information.

1. **Naïve Bayes is a linear classifier in the binary domain.**

YES, in a binary domain, naive Bayes classifier is a linear classifier, since our decision boundary is a hyperplane

1. **A Bayesian network represents a joint probability distribution.**

YES

1. **According to the principle of Occam’s razor, the simplest hypothesis should be chosen, regardless the performance.**

NO, based on this principle, in a similar performance we should choose the simplest hypothesis instead of complex one.

1. **AUC is an evaluation criterion that dominates accuracy (higher AUC guarantees higher accuracy).**

YES, higher AUC guarantees higher accuracy.

1. **Learning a decision tree can be done by a brute-force search in practice.**

NO, since our search space is extremely huge, we could not use brute-force.

1. **Bagging results in high variance.**

NO, Bagging is a technique for reducing the variance by means of voting.

1. **Perceptron cannot represent all the Boolean functions.**

YES, although it could represent some linear functions like AND, OR and NOT, it’s not able to represent nonlinear ones like XOR and XNOR.

1. **The structure of a Bayesian network represents conditional independencies among variables.**

YES

1. **When we build a multi-layer artificial neural network (ANN) and want to use gradient descent to learn an ANN, perceptron is a suitable neuron model.**

It depends on its function, if it has singular point, the answer is NO otherwise, if it is derivable, the answer is YES.

1. **Learning Bayes optimal classifiers is practical.**

NO, in practice, the Bayes Optimal Classifier is computationally expensive.

1. **The optimal classification is not always the classification given by the most probable hypothesis.**

Yes,

1. **Instance learning does not have the problem of curse of dimensionality.**

NO.

1. **In SVM, one key idea is to convert a nonlinear problem to a linear one by applying a mapping function. However, we do not need to carry out the mapping function explicitly.**

YES, SVM uses kernel functions to convert non-linear classification to linear classification by mapping into higher dimensional space. However, we don’t do it directly.

***Briefly* answer the following questions:**

**Question 1: (5 points) A decision tree learning algorithm is a process of partitioning training data. So, as it goes, the data associated with leaf nodes become smaller and smaller, which typically ends up with only a very few instances on a leaf. This problem is called the fragmentation problem. What is the consequence of the fragmentation problem? Is it unavoidable? How to handle it?**

*overfitting*

*Yes,*

*post-prune*

*Stop earlier.*

**Question 2: (5 points) When does overfitting happen in machine learning? Analyze the overfitting for decision tree learning and naïve Bayes. Which one is more serious? Why?**

*When the algorithm follows all of the samples in training set instead of patterns. For example, in decision tree we partition all data even noisy data in training set. In the decision tree it is more serios. But naïve based is high bias so it does not happen in variance.*

**Question 3: (5 points) Occam’s razor is an important principle in machine learning. What is the principle of Occam’s razor? Use decision tree learning as an example to show how to apply Occam’s razor.**

*If the performance of two hypotheses were similar, the simplest one should be selected rather than the complex one. For example, in decision tree, if we have two hypotheses, with similar error and one of them was complex, we should select the simplest hypothesis.*

**Question 4: (5 points) Briefly describe Random Forest (RF) algorithm. Graduate student *Quest* at UNB claims that RF is a good algorithm, because it is robust (not sensitive) to data noise, such as outliers. What do you think? Justify your answer.**

*yes, this algorithm uses bagging algorithm. Therefore, it is robust to noisy data and overfitting. The reason here is that each tree uses different hypothesis for training then we vote among each tree and return the highest class.*

**Question 5: (5 points) Does naive Bayes generate accurate probabilities? Why? Propose your solution for this issue.**

*NO, because in naïve Bayes we consider conditional dependencies which is not realistic. Also, we assume all features have Gaussian distribution, we use the proper model for*

**Question 6: (5 points) Variance and bias are called evil twins in machine learning. Think of learning Bayesian networks. Discuss how to design a good learning algorithm for Bayesian networks from the perspectives of variance and bias.**

*Thus, Bayesian belief networks provide an intermediate approach that is less constraining than the global assumption of conditional independence made by the naive Bayes classifier, but more tractable than avoiding conditional independence assumptions altogether.*

*how to find a tradeoff between bias and variance in learning Bayesian networks*

*Bias is the simplifying assumptions made by the model to make the target function easier to approximate. Variance is the amount that the estimate of the target function will change given different training data. Trade-off is tension between the error introduced by the bias and the variance.*

*We should consider limited dependencies and don’t ignore them, as in Bayasian network conditional independencies are a major point.*

**Question 7: (5 points) Why both perceptron and linear units are not suitable in building multilayer ANN? Why hidden units are often used?**

*Linear unit make a linear relation between input and the output. If function of perceptron does not have differentiate, we could not apply back propagation for training.*

*Hidden units are used for nonlinear functions. Because perceptron could solve only linear problems therefore, we combine them to solve non linear problems*

**Question 8: (5 points) When we use KNN, the Euclidean distance is often used. However, there are two issues for using it: (1) attributes have different scales, for example, body temperature and blood pressure. (2) Some attributes are not numeric, for example, color (red, white, etc.). How to deal with these two issues? Briefly present your solutions.**

*we can do normalization or standardization. So our data will be converted between 0 and 1 or a normal distribution with zero mean and unit variance.*

*We can set a number to each symbolic attribute Or using probability distance.*

**Question 9: (5 points) In some applications, labeling instances by human experts is expensive. So a typical story is that only a very few labeled instances are given while a large number of unlabeled instances are available. The corresponding learning task is called semi-supervised learning. Design an EM algorithm to extend the labels of labeled instances to the unlabeled instances, in which the base learner is naïve Bayes.**

*The EM algorithm is an iterative approach that estimate the label of each missing variables, by optimizing the parameters of the model. Then after converging we can use another classifier.*

**Question 1: (5 points) In machine learning, the representational power of a model reflects what target functions are representable by it. Then, it seems that the more representational power a model has, the better is the model. What do you think? Explain your answer.**

*I think it’s not correct, for example in the deep learning we could not represent the model for human while in most cases the results of it are better than ID3 which is a representational power.*

*The most common type of machine learning is to learn the mapping Y=f(X) to make predictions of Y for new X.*

*This is called predictive modeling or predictive analytics and our goal is to make the most accurate predictions possible. As such, we are not really interested in the shape and form of the function (f) that we are learning, only that it makes accurate predictions.*

*We could learn the mapping of Y=f(X) to learn more about the relationship in the data and this is called statistical inference. If this were the goal, we would use simpler methods and value understanding the learned model and form of (f) above making accurate predictions. When we learn a function (f) we are estimating its form from the data that we have available. As such, this estimate will have error. It will not be a perfect estimate for the underlying hypothetical best mapping from Y given X.*

*Much time in applied machine learning is spent attempting to improve the estimate of the underlying function and in term improve the performance of the predictions made by the model.*

**Question 2: (5 points) Briefly describe Adaboost algorithm. Graduate student *Quest* at UNB argues that Adaboost is not a good algorithm, since the weights of incorrectly classified examples increase after each iteration that results in fitting the outliers in data too much. What do you think?**

*It seems, it could be a good algorithm since it tries to correct the miss classes by increasing their weights. So, the next weak classifier could put more attention to the miss classified samples. But this algorithm also could learn the outlier. As a result, it’s not good approach for noisy data.*

*Boosting increase variance. Also, if we use a more strong classifier instead of weak classifier (stump) we suffer from overfitting.*

**Question 3: (5 points) Explain dimentionality in Adaboost.**

*The dimensionality of the data: We know that in general, we experience overfitting more in high dimensional spaces ("the curse of dimensionality"), and AdaBoost can also suffer in that respect, as it is simply a linear combination of classifiers which themselves suffer from the problem. Whether it is as prone as other classifiers is hard to determine.*

------------------------------------------------------------

*Note.*

*Bagging decreases overfitting while increase bias [parallel classifier] RF*

*Boosting decrease bias while increase overfitting [series classifier] Adaboost*

[*About bias*](https://downey.io/notes/omscs/cs7641/restriction-and-preference-bias-supervised-learning/#:~:text=Restriction%20bias%20is%20the%20representational,model%20is%20able%20to%20represent)*:*

*Vanishing, exploding:*

*In the training process, the gradient would be zero for the first layers if you have many layers. As a result, the first layers could not learn the problem.*

*Relu() function and tanh() function*

*Generalization*

*Exploding*

*Batch normalization*

*Initial value for weights*