MACHINE LEARNING I EXAM – MASTER IN BIG DATA ANALYTICS

JANUARY 2016

24 questions, 1 point each (except the two last ones). 2 hours (17:00-19:00)

1. **What is the difference between classification and regression?**

Classification uses discrete / categorical labels, while regression uses real labels (i.e. its aim is to predict numerical values)

2. **What is semi-supervised learning?**

Only the labels of a few instances are known in the training set.

3. **Let us suppose that a company that sells books through the Internet has a database where every instance represents a customer together with the list of books that s/he has bought in the past. The company wants to use this information in order to recommend books to possible buyers. Which Machine Learning task could be used to solve the problem of the company?**

Market basket-analysis. It is an unsupervised task, whose aim is to find relations between attributes. In this case, we are interested in learning rules of this kind: if the customer buys book A then it also buys book B, in order to make recommendations about book B.

Clustering could also be an option, if properly justified.

4. **Define, or give an example, of "instance-space"**

E.g. in a problem with two input attributes, instance-space is a mathematical space with two dimensions.

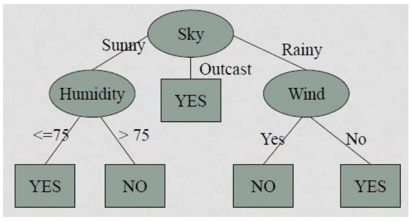
5. **In what kind of problems it is useful to use K>1 for the K-nearest neighbors algorithm and why?**

For instance, in problems where there is noise or class overlap. K>1 classifies instances as belonging to the majority class. It's a kind of averaging, which reduces the noise to some extent.

6. **What is the main limitation or inconvenience of KNN and why?**

There are several limitations, but the main one is that (1) it is necessary to store all training instances, and (2) in order to make a prediction, all distances to the training instances must be computed. Hence, if the training set is very large, it requires a large storage and computing predictions is very time consuming. Sensitivity to irrelevant attributes is another important hindrance.

7. **Write one IF-THEN rule from the following decision tree**



IF sky==sunny AND Humidity <=75 THEN Yes

8. **What is the difference between a regression tree and a model tree?**

Both are used for regression, but regression trees contain numbers in the leaves (they represent averages) and model trees contain linear models in the leaves.

9. **For classification decision trees, entropy is used in order to select the best attribute and to split a tree node. What is used in regression trees for this same purpose?**

Standard deviation or variance.

10. **Assume that a classifier is learned from some training data with three classes. Suppose that 25% of instances in the training data belong to the first class, 35% to the second class, and 40% to the third class. The error rate of this classifier on test data is 41.3%. Do you think that this classifier is useful? Why?**

If the error rate is 41.3%, the success rate is about 59%. In a problem with three classes, the success rate of random chance is 1/3 == 33%. The success rate of the trivial majority class classifier is 40%. Therefore, 59% is a useful result because it is larger than both random chance and the majority class classifier.

11. **Explain briefly the cross-validation method**

Cross-validation is a method for evaluating models. The available dataset is divided into k independent folds. The following process is repeated k times: train a model with all folds but fold i, test the model with fold i. Finally, compute the average of all classification rates. At the end, a definitive model is trained with the whole available dataset.

12. **Do we usually get better classifiers from cross-validation, compared to train/test? Why?**

Cross-validation (and train/test) is a method for evaluating models, not for building models. Therefore, it does not generate better (or worse) models, because that is not its purpose.

13. **Suppose that we want to train and test a linear model with two input attributes. Our available dataset contains 10^{12} instances. What would you prefer, train/test or cross-validation? Why?**

Cross-validation is typically used because single train and test partitions can be biased. However, if a very large dataset is available (as in this case), then the test set will be large enough to be representative of the problem. Moreover, linear models are very simple, especially with very few attributes, and the likelihood that they will overfit the biases contained in the data is small. Given that crossvalidation is computationally expensive, and that train/test in this case will produce a good estimation of the classification error , train/test should be preferred.

14. **What is the main limitation of PCA for classification problems?**

PCA is an unsupervised attribute transformation method (it does not take into account the labels). Therefore, for some problems, PCA with attribute selection might be misleading.

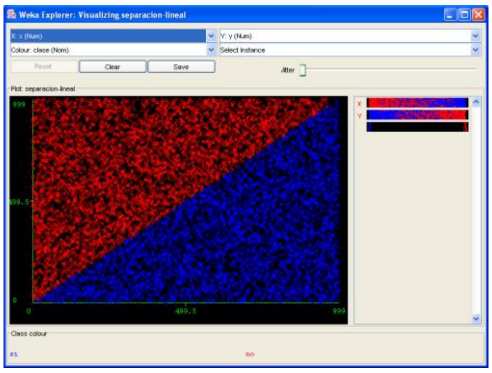
15. **Let us suppose KNN is used in a 2-dimensional problem, with a training set with 1500 instances and K = 1500. What would be the predicted class for an instance with input attributes (0.3, 4.7)?**

With K == number of training instances, KNN always returns the same class (the majority class).

16. **What kind of parallelism would you use to speed-up cross-validation: task parallelism or data parallelism? Why?**

Task parallelism, because all iterations of cross-validation must use (approximately) the same data. Therefore, data cannot be partitioned into non-overlapping partitions (at most, it would have to be replicated in all computers).

17. **Draw a picture of a 2-class 2-dimensional classification problem that would require a complex decision tree (i.e. containing many nodes) in order to approximate the boundary between positive instances and negative instances.**



A problem with classes separated with an oblique boundary would be difficult for a decision tree because boundaries in decision trees are either horizontal or vertical. The oblique boundary would have to be approximated with many horizontal or vertical boundaries and hence, would require many nodes in the decision tree, especially if the boundary is very long.

18. **Explain briefly what is *out-of-bag estimation* in Random Forests, and how it is computed**

Each tree in a Random forest is built from a sample generated by sampling with replacement. Some instances of the available data will be not be used for training some of the trees. Those instances not used for training can be used to compute the out-of-bag estimation, an estimation of the future performance of the RF. More specifically:

- For every instance I that belongs to the original data set, determine the subset of trees S in the RF for which instance I was not used for training the tree
- Classify I using trees S
- Compare the prediction with the actual class of I
- Add all successes and compute the success rate

19. **Let us suppose that there is a computer network with 1000 computers, each computer is able to process 500Gb of data in approximately 1 hour. Let us suppose that we execute in a Python notebook the following Pyspark sentence. How long, approximately, would it take to execute?**

    result_rdd = data_rdd.map(lambda x: 2*x)

    map is a transformation (not an action), and therefore by itself, it does nothing. This sentence would be almost instantaneous.

20. **Explain in what kind of problems Boosting might not work well and why.**

Problems with noise / class overlap. Boosting constructs new models by focusing on instances classified wrongly by previous models. Noisy instances are very hard to be classified correctly, if at all. Therefore, Boosting will focus on those instances and might end up memorizing them (overfitting).

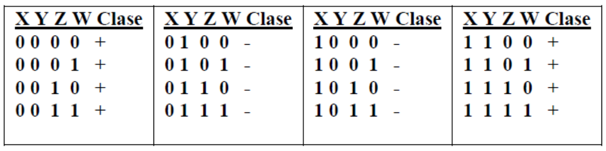
21. **Can a standard ranking attribute selection method detect redundant attributes? Why?**

No. If two redundant attributes are very correlated with the class, they will both be ranked highly, and therefore, they will likely be selected together.

22. **Give two reasons for why attribute selection can be important in Machine Learning.**

- It removes irrelevant attributes and this can improve the quality of the model
- It removes redundant attributes and this can make the model simpler
- It reduces the curse of dimensionality

23. **(2 points)** What attribute selection method would you use with the following two-class dataset, and why? Note: there are four input attributes (X,Y,Z,W) and the class (“clase”)



X	Y	Z	W	Clase
0	0	0	0	+
0	0	0	1	+
0	0	1	0	+
0	0	1	1	+
0	1	0	0	-
0	1	0	1	-
0	1	1	0	-
0	1	1	1	-
1	0	0	0	-
1	0	0	1	-
1	0	1	0	-
1	0	1	1	-
1	1	0	0	+
1	1	0	1	+
1	1	1	0	+
1	1	1	1	+

Attributes X and Y, taken in isolation, are not correlated with the class. E.g. if X=0, the class can be + or -. If X==1, the class can be + or -. Similarly for Y. But taken together, they predict the class exactly (XY==00 => class +, XY=01 => class -, etc.). So, we would need an attribute selection method that is able to evaluate subsets of attributes and take into account interactions between them (like Wrapper). CFS would not be useful in this case, because although it can penalize redundant subsets, it evaluates correlations of attributes with the class individually.

24. (2 points) Can the process of training a decision tree be parallelized in Mapreduce by learning different subtrees in different computers? Discuss why this could be (or not) a good idea.

It would not be a good idea, because Mapreduce requires data to be splitted into independent partitions in different computers. But two subtrees of the same tree would require, in principle, the same dataset. It is true that once an attribute has been selected for a node, data is partitioned into the different branches, but we can only know how to partition the dataset AFTER the attribute has been selected. But in MapReduce, data must be partitioned BEFORE the algorithm starts (and therefore, before an attribute is selected).