Project 2 - DMML 17/18

by Yimin Xie, Yantao Shi, Jeong-Eun Choi

TECHNISCHE UNIVERSITÄT DARMSTADT

Ex.1 - Decision Trees

TECHNISCHE UNIVERSITÄT DARMSTADT

- Goal: compare J48 and ID3
- Condition:
  - Two Classification Datset: diabetes.arff and ionosphere.arff
    - with only two classes in order to compare the ROC curves (otherwise need to compare all the curves of each classes)
  - J48 with and without pruning (all other values to default)
  - ID3 (all values to default)
  - Use 10x10 CrossValidation (randomness to achieve better result)
- Problem: ID3 only takes nomial values
  - use filter:filters.supervised.attribute.Discretize-Rfirst-last-precision6
  - additional question: need to use the filtered dataset for J48 and ID3?
    - we used for both J48 and ID3 filtered data so that we have fixed data to compare with

Ex.1 - Prediction

TECHNISCHE UNIVERSITÄT DARMSTADT

- **ID3**: The implementation in weka is constructing an unpruned decision tree based on the ID3 algorithm. The core is to apply information gain criteria selection features on each sub-node of the decision tree to construct the decision tree recursively. However, it can not handle missing values and requires nominal attributes to compute. It uses a heuristic for choosing an attribute - attribute with the highest information gain is chosen
- **J48**: The implementation in weka is generating a pruned or unpruned C4.5 decision tree. C4.5 is an improved algorithm of ID3 that can handle missing values and does NOT require nominal attributes to compute. Furthermore, it uses a heuristic for choosing the attribute - it uses the normalized information gain that results from choosing an attribute for splitting the data. The attribute with the highest normalized information gain is chosen

- **Prediction**: since J48 can prune the tree, we expect that the **tree of J48 is much smaller** than the one from ID3. Moreover, we expect that the heuristic of choosing the attribute with normalized information gain is better than pure information gain. Therefore, even if we use **J48 unpruned, the tree will be smaller** than the one from ID3. Since we are doing **10x10 cross-validation**, the accuracy of J48 will be better since it prevents overfitting through pruning. In case of **training set**, the accuracy of ID3 will be higher, since it aims to maximize the accuracy to the training data given, while J48 prunes, which would result lower accuracy.

Source: A comparative study of decision tree ID3 and C4.5 (link)

Ex.1 a) - ROC Curves

TECHNISCHE UNIVERSITÄT DARMSTADT

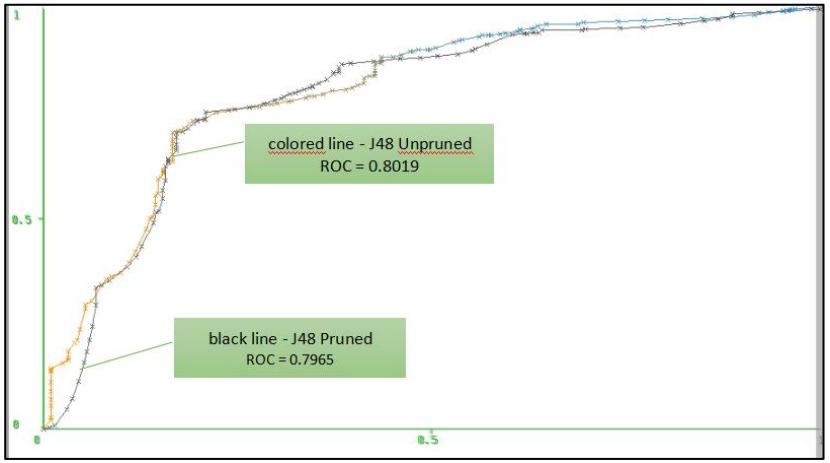

Fig1. diabetes J48 pruned vs. unpruned

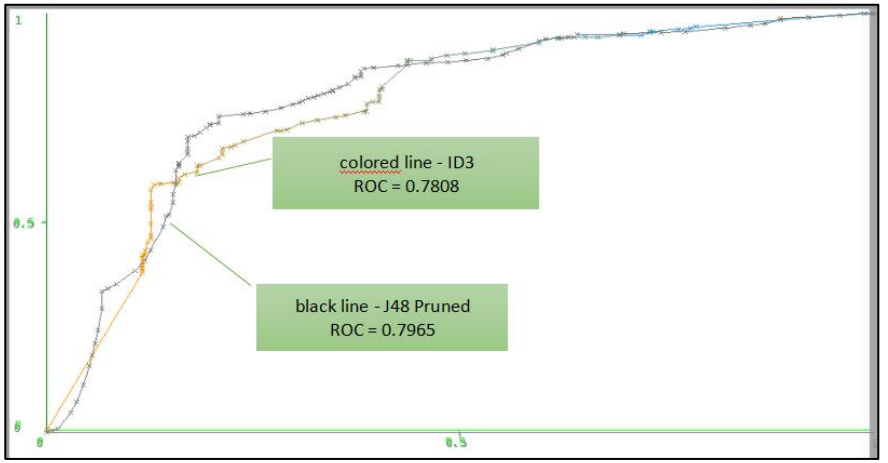

Fig2. diabetes J48 pruned vs. ID3

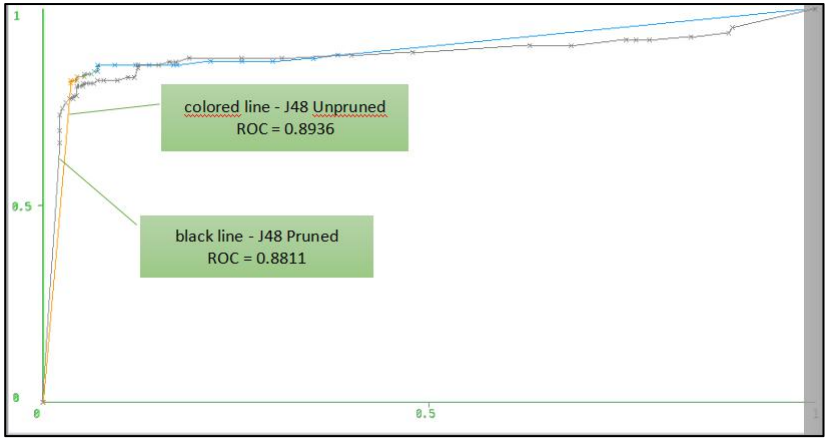

Fig3. ionosphere J48 pruned vs. unpruned

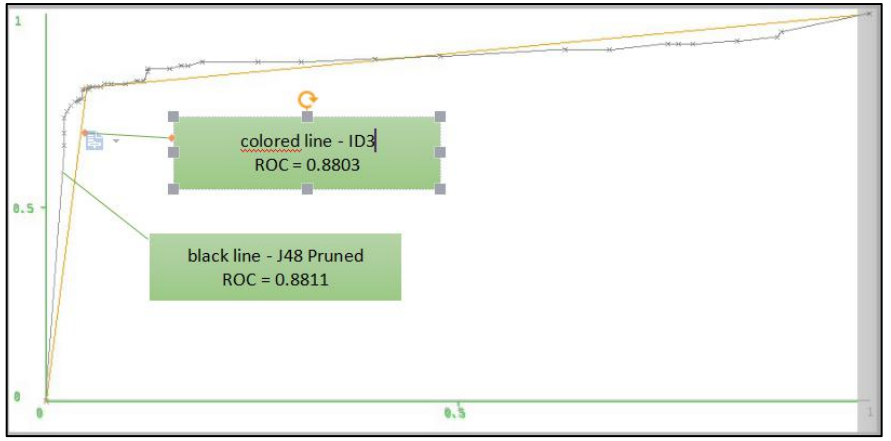

Fig4. ionosphere J48 pruned vs. ID3

Ex.1 b&c) - Accuracy and Tree

TECHNISCHE UNIVERSITÄT DARMSTADT

10x10 CV	Accuracy (%)	#Nodes	#Leaves
ionosphere (J48 unpruned)	90.3134	57	44
ionosphere (J48 pruned)	89.4587	27	21
ionosphere (ID3)	88.8889	89	71
diabetes (J48 unpruned)	76.0417	54	31
diabetes (J48 pruned)	77.7344	22	13
diabetes(ID3)	76.5625	173	94

Fig5. J48 & ID3 - 10x10CV

TRAININGSET	Accuracy (%)	#Nodes	#Leaves
ionosphere (J48 unpruned)	98.2906	57	44
ionosphere (J48 pruned)	95.4416	27	21
ionosphere (ID3)	**100**	89	71

Fig6. J48 & ID3 - TrainingSet

Ex.1 Explanation & Conclusion

TECHNISCHE UNIVERSITÄT DARMSTADT

- For this experiment, we always took the class of “recurrence events” to generate ROC-curves. X-axis stands for 'false positive rate' and Y-axis for 'true positive rate'. So, the area under ROC-curve also represents the **accuracy** of the algorithm used. The results from J48 unpruned always achieved the biggest area under ROC curve, which matches to our expectation that J48 would reach the higher accuracy than ID3. It is also interesting to compare the curves between pruned and unpruned J48, since you can see how pruning affected the curves (Fig1. and Fig3.). In ID3 we can observe that parts of the curves are quite linear (Fig2. and Fig4.), which might be a **reflection of the algorithm** ID3. Moreover, we can learn about the **dataset**. For example, 'ionosphere' can reach quite high rate of true positive rate with relatively low false positive rate(Fig3. and Fig4.). So, this represents that this dataset is already quite *suitable* (and has suitable attributes) to classify, while for 'diabetes' one might include some other attirbutes (or weight each attribute differently) in order to make it more *suitable* to classify.
- As expected, the trees of **ID3 is much bigger** than those of J48 either pruned or unpruned.(Fig.5) Of course, pruned J48 generated much smaller tree and lost only small range of accuracy. In case of 'diabetes' it actually reached highest accuracy, which might be a proof of 'preventing overfitting'. When we observe the results of **'training set'**(Fig.6), we see that ID3 reached 100% of accuracy. So, ID3 creates a 'full-tree' for a training set it uses which leads to overfitting. It is also interesting to see that once the tree is generated that the trees are NOT being updated which is a important chracteristics of ID3 and J48 (Fig5. and Fig6.)
- As a conclusion, ID3 has the tendency of overfitting and creates too large tree. J48 even without pruning generates smaller trees due to better heristics of choosing attributes. However, even if we prune the tree we do not loose a lot of accuracy and sometime gain from 'preventing overfitting' and creates the smallest trees. **So, J48 pruned appears to be the best to use.**

Ex.2 - NearestNeighbor, k-NN

TECHNISCHE UNIVERSITÄT DARMSTADT

- Goal: test and find which k reaches highest accuracy by using k-NN
- Condition:
  - use k-NN (IBk, with default parameters)
    - while $k \in \{1,3,5,7,9,11\}$
  - use the same dataset from the previous Exercise 1 with the same filter
  - 10x10 Cross Validation

Ex.2 - Prediction

TECHNISCHE UNIVERSITÄT DARMSTADT

- The **K-NN** uses aTraining Set to predict the value of a variable of interest for each member of a target data set (lazy approach). An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. k represents a certain 'distance' like Euclidian distance.
- **Prediction**: If k = 1, then it gurantees an error rate of no worse than twice the minimum achievable error rate given the distirbution of the data. So, k = 1 is more like our baseline. However, if k is too big, this would increase the probabilty of taking unsuitable data into account. Therefore, we assume that **k =3 or 5** would be the best k-value, since our datasets are not very big. 'diabetes' has 768 instances and only 9 attributes. 'ionosphere' has only 351 instance and 35 attributes.

Ex.2 - Results & Explanation

TECHNISCHE UNIVERSITÄT DARMSTADT

	k = 1	k = 3	k = 5	k = 7	k = 9	k = 11
#ionosphere	94.0171	90.5983	90.0285	89.7436	89.4587	89.1738
#diabetes	77.0833	76.6927	76.8229	76.8229	75.9115	75.2604

Fig7. k-NN results

- Different from our expectation, we could achieve the best results from k = 1. Theoretically, if the number of samples is infinite, the bigger the k, the better. In our experment, it appears that the dataset is small so that k = 1 is already good enough and the algorithm is not very sensitive to noise (or there is not much noise in the dataset).

Ex.3 - Regression Trees

TECHNISCHE UNIVERSITÄT DARMSTADT

- Goal: Analyze the results of M5P by testing it on five regression datasets
- Condition:
  - using datasets:auto-price, concrete, housing, stock, wine-quality (no filter used)
  - using M5P with and without pruning (all other values to default)
  - using M5P with regression tree and model tree
  - 10 Cross Validation (as instructed)

Ex.3 - Prediction

TECHNISCHE UNIVERSITÄT DARMSTADT

- **M5**: In weka the M5P implementes base routines for generating M5 Model trees and rules. It can also create a regression tree. M5 "builds tree-based models but, whereas regression trees have values at their leaves, the tres constructed by M5 can have multivariate linear models; these model trees are thus analogous to piecewise linear functions. [...] The advantage of M5 over CART is that model trees are generally much smaller than regresstion trees."
- **Regression Trees** just have constant fitted mean of the response in each node which is then used for the prediction. **Model Trees** can fit a regression model within each node of the tree. I.e. it can set the leaf nodes as piecewise linear functions, where piecewise linear means The model consists of multiple linear segments, which is the model tree. The interpretability of the model tree is one of its characteristics superior to the regression tree. In addition, the model tree also has a more accurate prediction accuracy.
- There are various of error fuctions that can be used for regression tasks like MAE (Mean Absolute Error) and RMSE (Root Mean Squared Error) **MAE** is the average magnitude of the errors in a set of predictions without weighting the errors of each individuals. **RMSE** also measure the magnitude of error but it is the square root of the average of squared differences between prediction and actual observation. Both measurements are indifferent to the direction of erros and lowever values are the better ones. By squaring the errors before they are averaged, it gives higher weight to large errors. Therefore, RMSE is always bigger or equal to MAE, while if MAE = RMSE, all errors have the same magnitude.
- **Prediction**:Overall **Model Trees will achieve better accuracy** then Regression Trees. The RMSE will always be larger or equal to MAE, however, we predicted that with **pruning the RMSE might be lower**, since pruning might prevent overfittig effect i.e. smaller magnitude of errors.

Source: Learning with Continous Classes (link)
Source: MAE and RMSE - Which Metric is Better? (link)

Ex.3 Results & Explanation

TECHNISCHE UNIVERSITÄT DARMSTADT

	with pruning				without pruning			
MeanAverageError(MAE) /RootMeanSquaredError(RMSE)	MAE		RMSE		MAE		RMSE	
RegressionTree(RT) /ModelTree(MT)	RT	MT	RT	MT	RT	MT	RT	MT
#1auto-price	2096.3675	1466.5565	3336.3692	2171.1561	2075.0678	1403.2007	3287.1186	2094.5903
#2concrete	6.7866	4.7397	8.6751	6.3652	6.4819	4.2748	8.3325	5.8917
#3housing	3.2864	2.5047	4.8185	3.7502	3.1955	2.385	4.7203	3.7105
#4stock	1.1874	0.6707	1.6019	0.9429	1.1731	0.6656	1.5874	0.9274
#5winequality	0.5549	0.5484	0.7211	0.7092	0.5325	0.5147	0.6983	0.6811

- As expected, the Model Tree achieves always better results then Regression Trees. Considering accuracy only without pruning achieved better results. In our experiment accuracy gained by not pruning is higher than can be gained through pruning (by preventing overfitting amd decreasing large errors). However,whether to choose with pruning instead or without depends on the purpose of the regression task. For example when we are predicting auto-prices, large errors on guessing the prices might not be particularly important to consider (bargaining for final price). One might therefore use with pruning so that less resource is required for the tree. For stock or for winequality, one might favor more accuracy and lower large errors, then one would chosse without pruning instead.