IST707 Data Analytics

HW6: naïve Bayes and decision tree for handwriting recognition

Introduction:

We are trying to compare two classification algorithms, decision tree and naïve Bayes and choose the best algorithms for handwriting recognition. The goal is to recognize digits 0 to 9 in handwriting images. The data set comes from the Kaggle Digit Recognizer competition (https://www.kaggle.com/c/digit-recognizer/data). Because the original data set is so large, it has been sampled to several smaller set and got loaded into Orange. Then we used data to construct prediction models using naïve Bayes and decision tree algorithms. The parameters have got tuned to get the best model according to cross validation. Then result has got successfully submitted to Kaggle.

1- Data preprocessing steps:

   a. First the pixels with the maximum value of 0 has got removed. These pixels are always empty, and we try to remove as many as useless features as possible to avoid the curse of dimensionality.

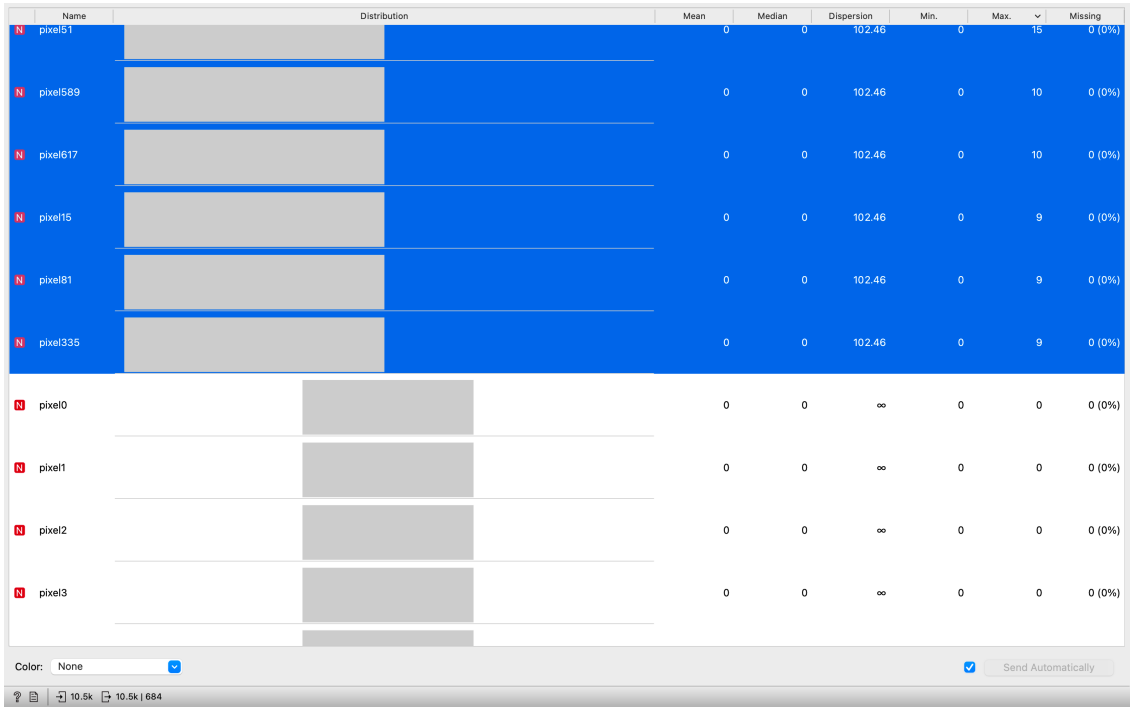


Figure 1-a: Selecting Features with Feature Statistics Module

b. The column containing label values has got selected as target, index has got selected as meta data, and all the pixels as features.

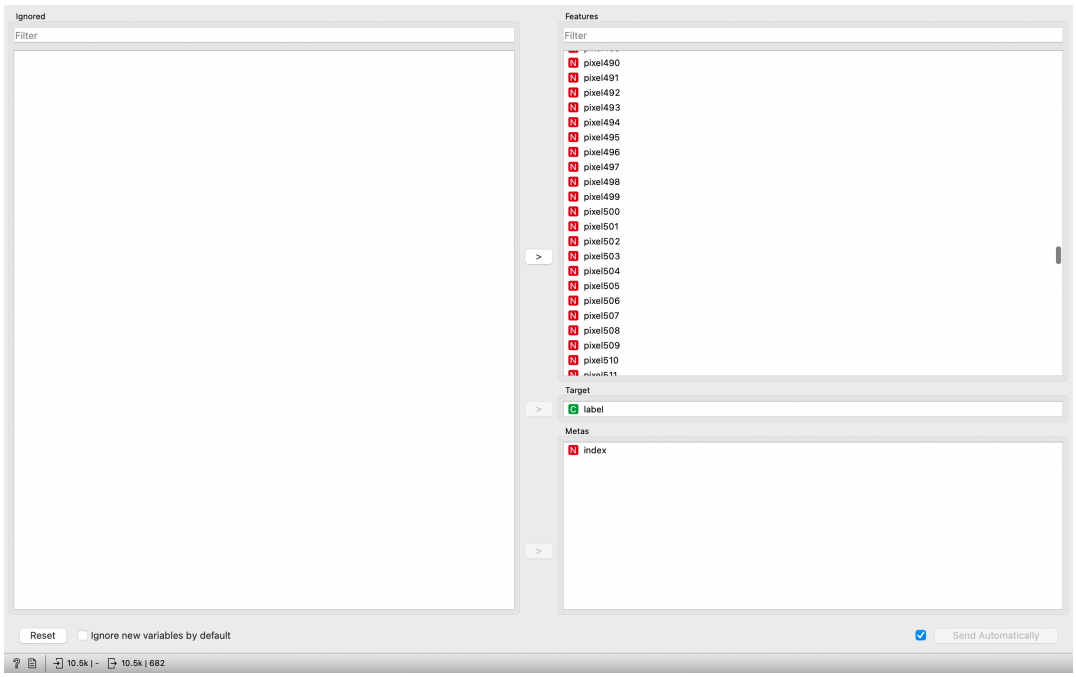


Figure 1-b: Selecting Columns with Select Columns Module

c. Impute module is used to eliminate rows with missing values. We can see that there are no rows with missing values in this data set because input and output of this module have the same number of observations.

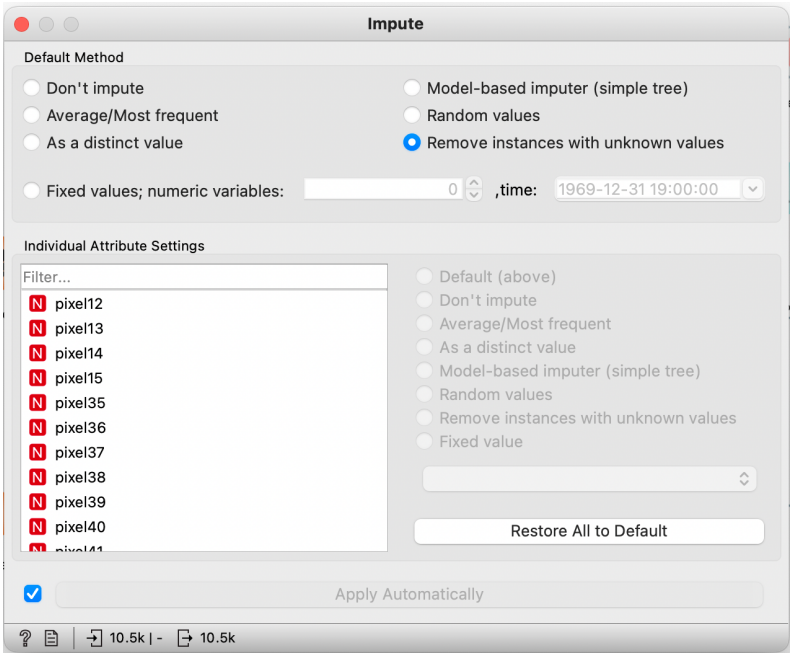


Figure 1-c: Eliminating Redundant Rows with Impute Module

d. After experimenting all the ranking combination, best accuracy got achieved for both models while ranking the highest 602 features regarding FCBF scoring method.

Figure 1-d: Selecting Top 602 Features with Regard to FCBF scores.

e. After experimenting all the discretization methods, best accuracy got achieved for both models while using Entropy-MDL Discretization.

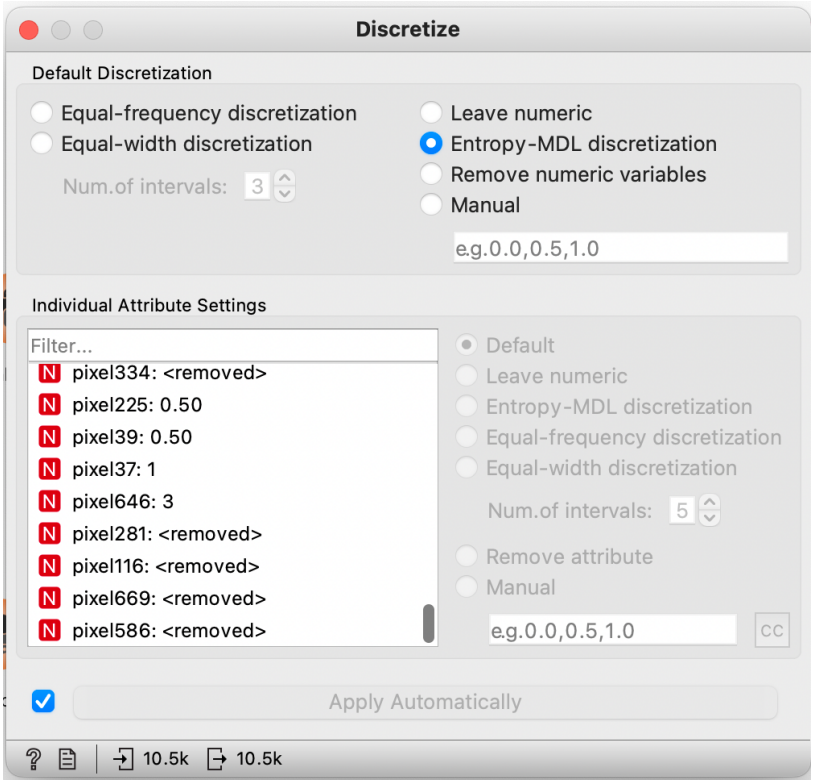


Figure 1-e: Discretizing the Features Using Entropy-MDL Discretization.

f. After experimenting all the sampling proportions, best accuracy got achieved for both models while sampling 21 subset using Cross Validation.

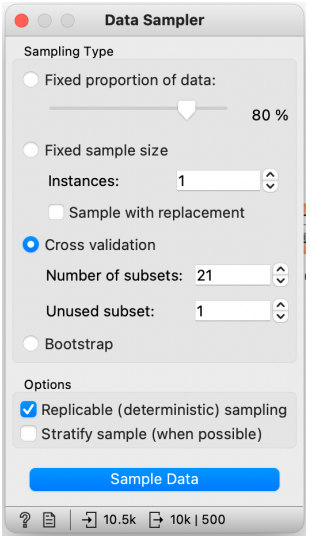


Figure 1-f: Using Cross Validation to Sample 21 Subsets.

2- Decision tree:

a. A decision tree has been constructed. After tuning the parameters multiple times, the best 3-fold cross validation accuracy that has been achieved when binary tree was induced, minimum number of instances in the leaves was 2, subset smaller than 4 was not splatted, the tree depth was limited to 9, and we stopped when majority reached 95%.

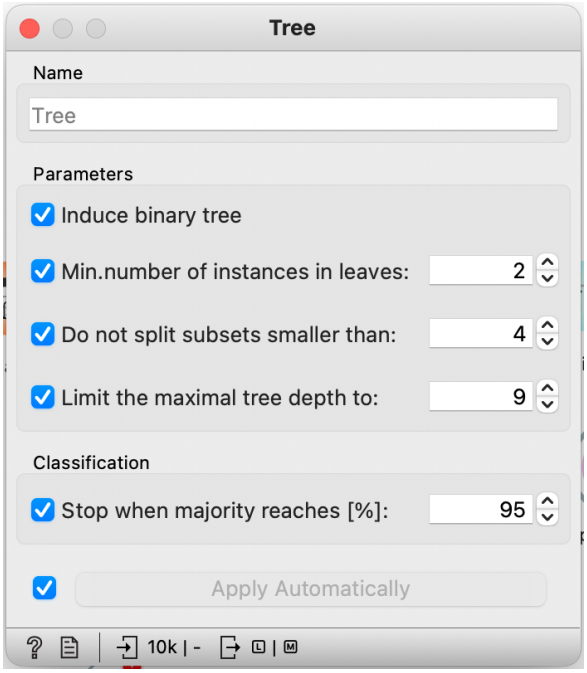
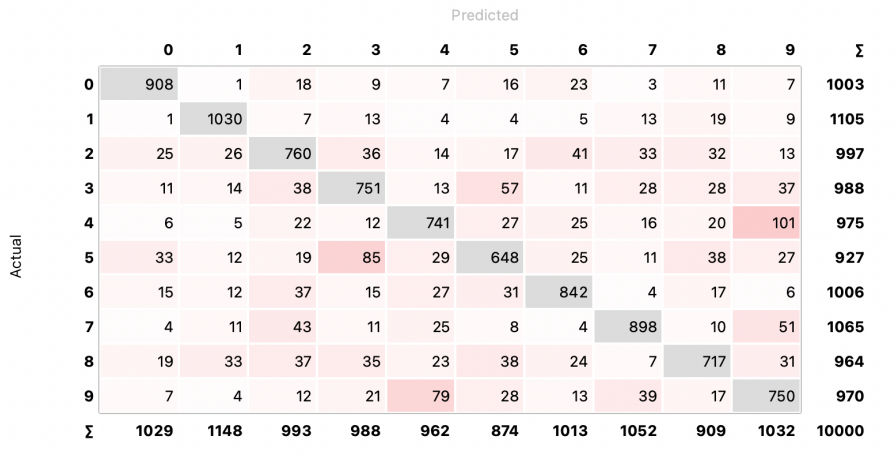


Figure 2-a: Decision Tree Configurations.

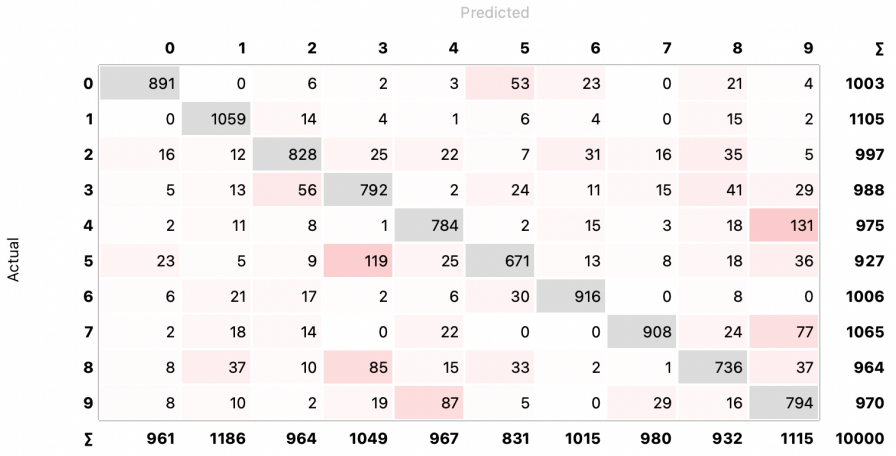


	0	1	2	3	4	5	6	7	8	9	Σ
0	908	1	18	9	7	16	23	3	11	7	**1003**
1	1	1030	7	13	4	4	5	13	19	9	**1105**
2	25	26	760	36	14	17	41	33	32	13	**997**
3	11	14	38	751	13	57	11	28	28	37	**988**
4	6	5	22	12	741	27	25	16	20	101	**975**
5	33	12	19	85	29	648	25	11	38	27	**927**
6	15	12	37	15	27	31	842	4	17	6	**1006**
7	4	11	43	11	25	8	4	898	10	51	**1065**
8	19	33	37	35	23	38	24	7	717	31	**964**
9	7	4	12	21	79	28	13	39	17	750	**970**
Σ	**1029**	**1148**	**993**	**988**	**962**	**874**	**1013**	**1052**	**909**	**1032**	**10000**

Figure 2-b: Confusion Matrix for Decision Tree.

3- Naïve Bayes

a. A Naïve Bayes model was also built, and the parameters has got tuned for best results.



	0	1	2	3	4	5	6	7	8	9	Σ
0	891	0	6	2	3	53	23	0	21	4	**1003**
1	0	1059	14	4	1	6	4	0	15	2	**1105**
2	16	12	828	25	22	7	31	16	35	5	**997**
3	5	13	56	792	2	24	11	15	41	29	**988**
4	2	11	8	1	784	2	15	3	18	131	**975**
5	23	5	9	119	25	671	13	8	18	36	**927**
6	6	21	17	2	6	30	916	0	8	0	**1006**
7	2	18	14	0	22	0	0	908	24	77	**1065**
8	8	37	10	85	15	33	2	1	736	37	**964**
9	8	10	2	19	87	5	0	29	16	794	**970**
Σ	**961**	**1186**	**964**	**1049**	**967**	**831**	**1015**	**980**	**932**	**1115**	**10000**

Figure 3-a: Confusion Matrix for Naïve Bayes.

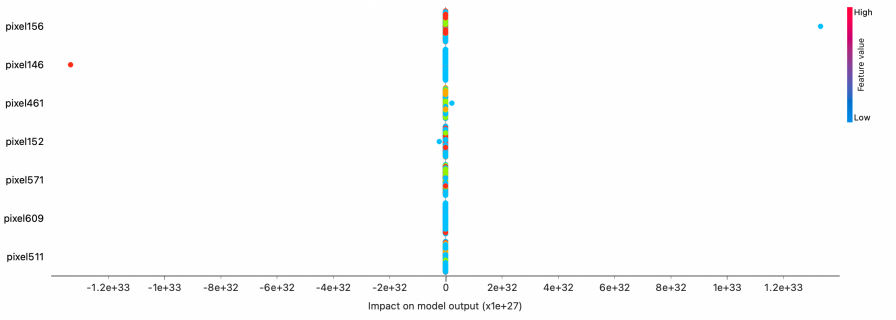


Figure 3-b: Impact of Features on Model Output for Target Value of 0.

4- Algorithm performance comparison:

While we got a better accuracy with Naïve Bayes algorithm, Decision Tree was faster. As there are no calculations involved in Decision Tree's classification process, it can classify data quickly. Classification that adheres to tree rules is quicker than that which requires calculation, such as in the case of Naive Bayes.

To compare the accuracy, cross validation with 3 folds was used. We can see that Naïve Bayes algorithm with accuracy of 0.84 outperform Decision Tree with accuracy of 0.80. Furthermore, the area under the curve is 98 percent for Naïve Bayes compared to 90 percent for Decision Tree. Thus, the best algorithm for handwriting recognition is Naïve Bayes.

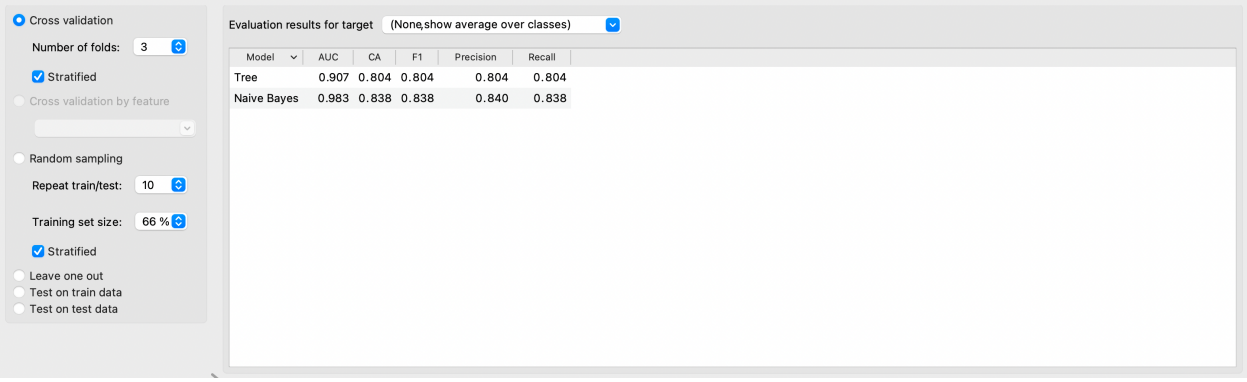


Figure 4-a: Test and Score Results for Both Naïve Bayes and Decision Tree Algorithms.

5- Kaggle test result

After building and tuning the models, prediction module was used to predict the numbers in the test data set provided by Kaggle. Then the CSV file was saved as Results.csv. If the accuracy provided by Kaggle website is less than what we see in the test and result output, we could say that the model is overfit because the model performed well on the training data but did not perform well on the evaluation data.

	ImageId	Label
1	15274	6
2	23603	9
3	12325	7
4	1812	0
5	10403	1
6	17474	7
7	464	5
8	13524	4

Figure 5-a: Prediction data table.