CITY
UNIVERSITY OF LONDON
EST 1894

A comparison between Random Forest and Decision Tree on MINST dataset

Mohsen Salimy

Description and motivation:

- Compare and contrast the performance of Random forest and Decision Tree on the MINST dataset to predict and learn the technique and pattern recognition on real world data. [1]
- This MINST data set includes handwritten digits centred in a fixed image originated from larger master set NIST.
- We will also compare how the digit image MINST was originally selected by Chris Burges and Corinna Cortes and how it the new version provided by Yann Lecun's.

Initial Analysis of Dataset

- Dataset: MINST database of handwritten digits
- Training set contains 60,000 examples and a test set of 10,000 examples
- The handwritten digits are size normalised and centred in a fixed image
- Originally the images were 20x20 pixel box in grey level image because of anti-aliasing
- Now the image are 28x28 due to computation of centre of the images. [1]
- The Images are around 30,000 in both SD-1 and SD=3 respectively, this is the when the training is set.
- They are many tested with a full sample of 60,000 training sets

Pros & Cons of the two Machine learning algorithms

Random Forest (RF)

- Applied to build a massive forest of decision trees to figure out the class for classification problems
- Can be used to ensemble algorithm in that forms the decision trees
- Utilised in hyperparameters to extract a large tree, features, depth and leaf's.
- Can be used to produce models by making use bootstrap process.

Pros: Put together to be applied for regression and classification problems

- Exercised to be used well for large datasets and it can cope well with superior dimensions
- Due to its opposing overfitting data it can be unaffected by imbalance data or noise.
- This classification has got towering accuracy performance compare to it's pairs.

Cons: Big deep trees causes longer wait time and expensive computation

- Can get confused in evaluation because of it's large tree

Hypothesis Statement

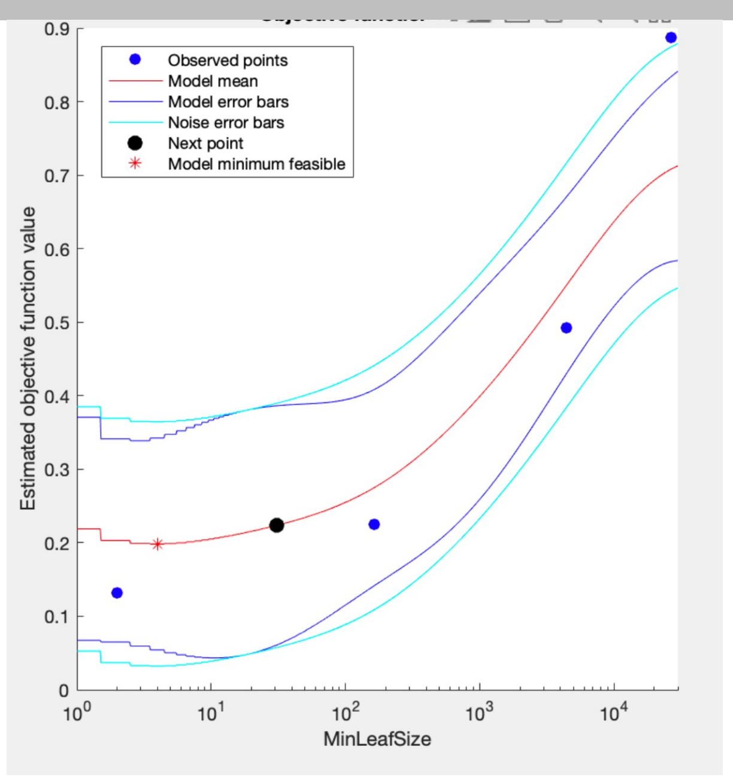
- Both Algorithm expected to perfume relatively well due to it's over lapping to each other in accuracy, test, train error loss
- According to [3] random forest has a higher training time and interpreting the data is not a concern where is for decision tree it is much easier to work it out and they are very fast with less compute power required.
- Reported by [4] that the redistribution error rate of RF is less then DS but it took more time to execute the data - this finding was experimented in medicine.
- Another experiment was done in MINST dataset from [5] DS had 85% accuracy and RF 94%.
- Although overall both are not accurate and productive for imbalance data although RF have an edge.

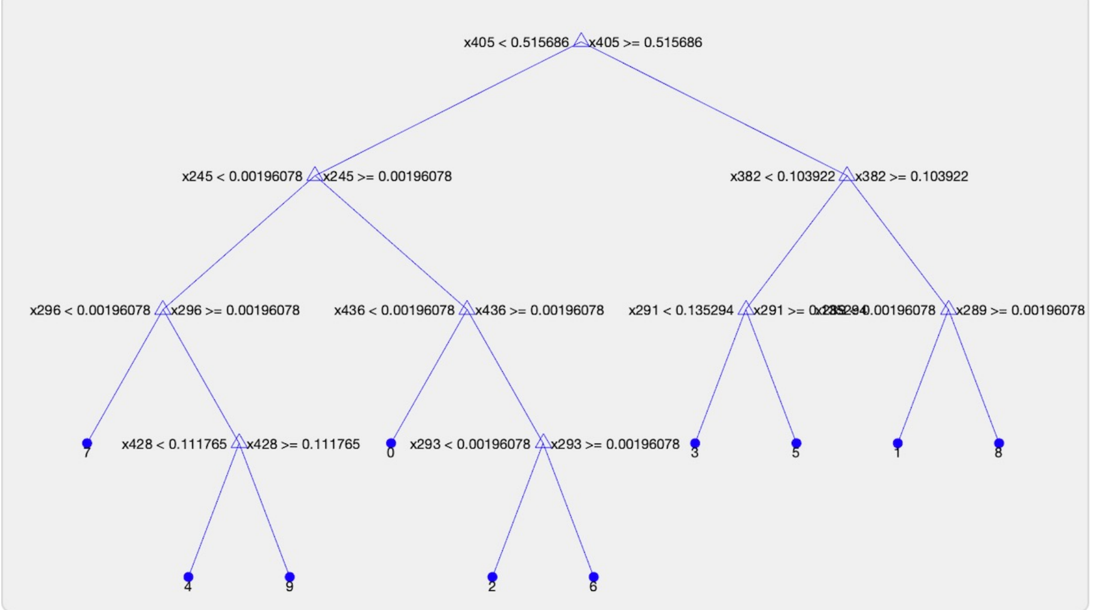
Choice of Parameters and experimental results

Random Forests : Parameters - Hyperparameters was used including max feature and min sample leafs. The trees ensembled were optimised using classification loss error training to obtain a good accuracy . Also run a confusion matrix heat map including deep trees on the two hyperparameters.

Result - The confusion matrix showed a good a level of predication and accuracy. It was done before and after hyperparameter optimization to see the differences

- The test error rate was 0.029555 this is quite accurate and it shows that there is one data points in the terminal nodes and so the trees in the forest has got good predication of data points.
- Also from the confusion matrix you could tell that classifier prediction was correct and so therefore it predicated all classes .
- Due to deep trees the time it took was longer then expected and it required a lot of computation power to run all those trees.
- The performance was inconstant at first however it got better as it went along the training
- The estimated objective function value is 0.15601 where is the function evolution time was 1805.28





Decision tree

- Decision Trees (DTs) are used for classification and regression problems by using it's data features and learns from it to make complex decision. [2]
- DT are supervised problems which makes them to have various of nodes to make the tree structure. [2]

Pros: The cost of using tree in the problems are used as data points to train the tree which makes very unique and simple

- Can able to handle both numerical and categorical data
- It can do well even the expectations are ruined by the exact model.
- Can be used a black box and be explained by Boolean logic arithmetic operations.
- Human can be able to read clearly and read by the tree displayed

Cons: Overfitting which can be complex which would require more leaf node to do well.

- The trees can sometimes be not clear due to the data.
- Sometimes it can create biased trees due to dominate classes.
- It can be susceptible to outliers and inclined to continue variables.[2]

Methodology

To meet the objective of project a comparison of the two algorithms was made taking into account the fairness and time for the two algorithms to be trained and tested to produce best models.

- Loaded and reshaped the data ,Fit data, train, test , predict both RF and DT
- created a visualisation of the tree classification, Produce heatmap ie confusion matrix for RF and DS, Used Bayesian optimization to implement hyperparameter tunning . Created and trained the best models when the model was optimized then implemented another confusion matrix for for the optimized models than predicted the optimized algorithms
- And overall models were compared

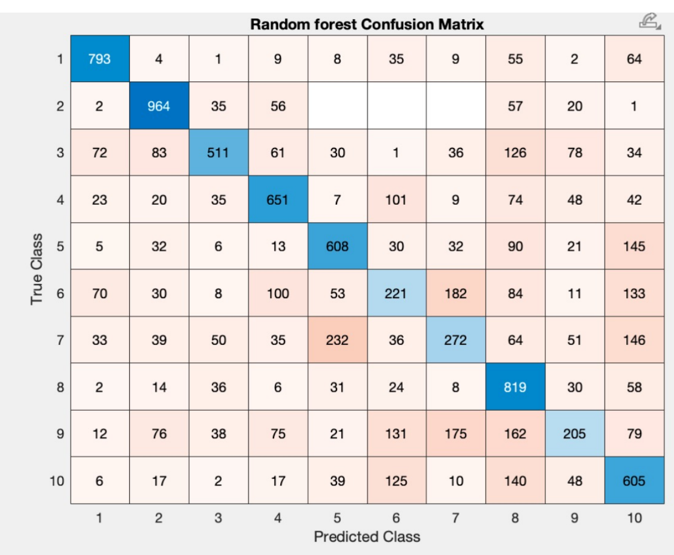
Decision Trees

Parameters : confusion matrix, tree classification, test and train errors.

- optimized the best trained models as well as optimized confusion matrix. Use of best observed feasible point – MinLeafSize,

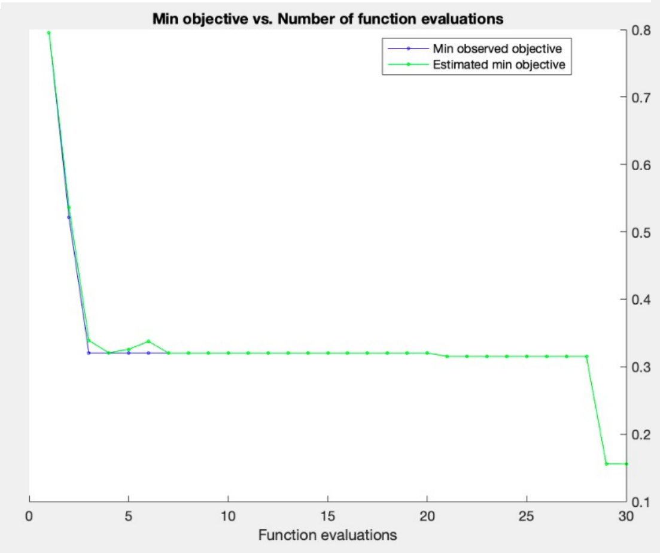
Results: After test error we obtained an accuracy of 0.43581 and train error: 0.44182.

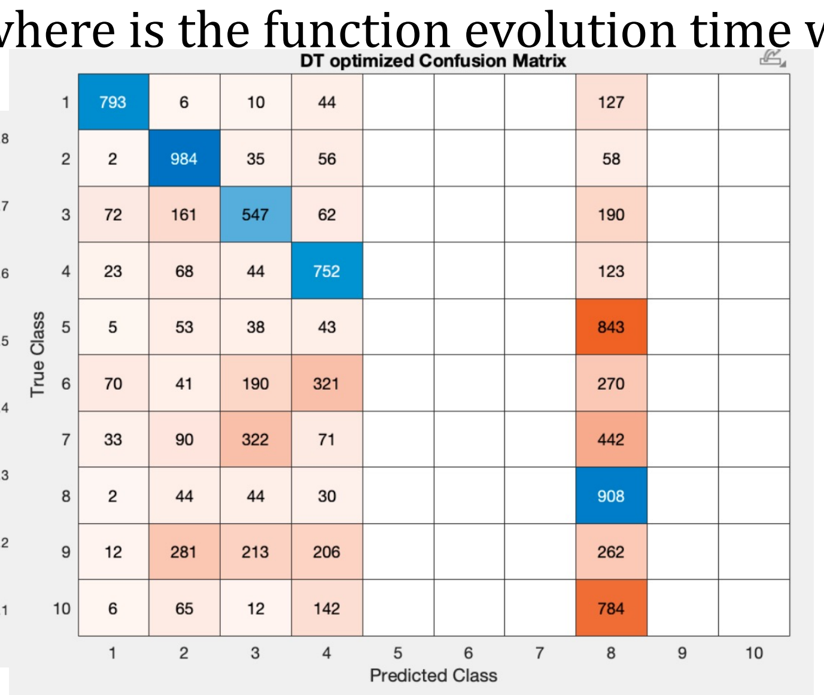
- The total time it took was 1202.4895 seconds to reach 30 max0bjectivevualations with minLeafSizr 7 and Maxnumsplits 4. This made it not complex or overfit.


Random forest Confusion Matrix

True Class	1	2	3	4	5	6	7	8	9	10
1	793	4	1	9	8	35	9	55	2	64
2	2	964	35	56				57	20	1
3	72	83	511	61	30	1	36	126	78	34
4	23	20	35	651	7	101	9	74	48	42
5	5	32	6	13	608	30	32	90	21	145
6	70	30	8	100	53	221	182	84	11	133
7	33	39	50	35	232	36	272	64	51	146
8	2	14	36	6	31	24	8	819	30	58
9	12	76	38	75	21	131	175	162	205	79
10	6	17	2	17	39	125	10	140	48	605

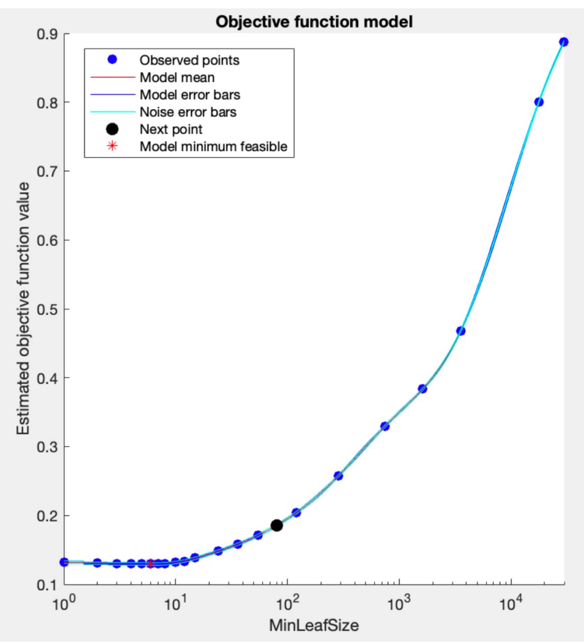
Predicted Class

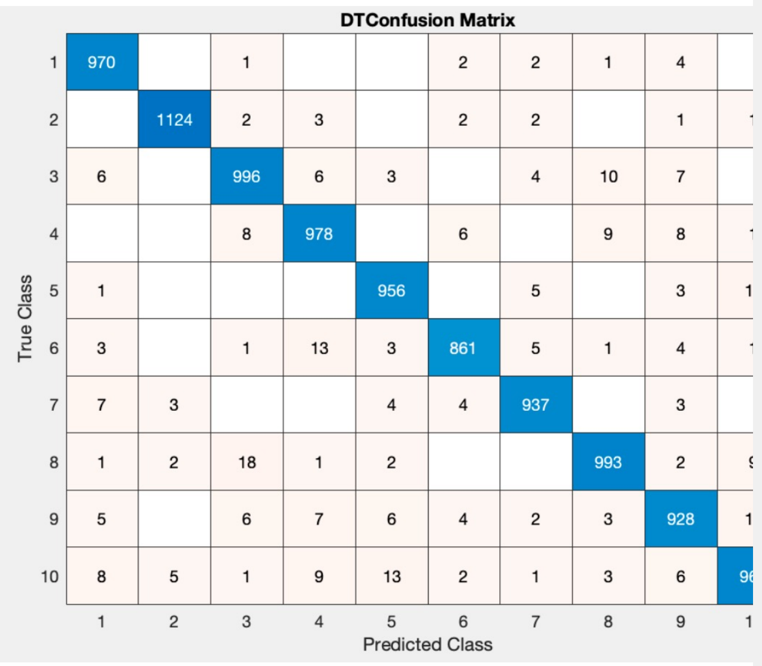



DT optimized Confusion Matrix

True Class	1	2	3	4	5	6	7	8	9	10
1	793	6	10	44				127		
2	2	984	35	56				58		
3	72	161	547	62				190		
4	23	68	44	752				123		
5	5	53	38	43				843		
6	70	41	190	321				270		
7	33	90	322	71				442		
8	2	44	44	30				908		
9	12	281	213	206				262		
10	6	65	12	142				784		

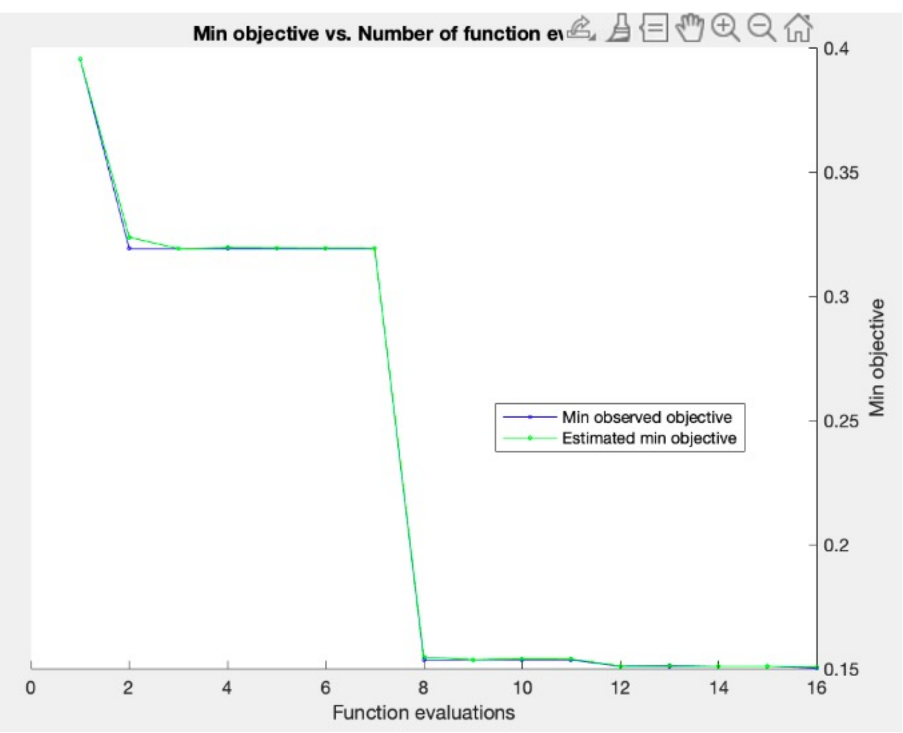
Predicted Class




DTConfusion Matrix

True Class	1	2	3	4	5	6	7	8	9
1	970		1			2	2	1	4
2		1124	2	3		2	2		1
3	6		996	6	3		4	10	7
4			8	978		6		9	8
5	1				956		5		3
6	3		1	13	3	861	5	1	4
7	7	3			4	4	937		3
8	1	2	18	1	2			993	2
9	5		6	7	6	4	2	3	928
10	8	5	1	9	13	2	1	3	6

Predicted Class



ANALYSIS AND CRITICAL EVALUATION OF RESULT

- The complexity of the data used and the various deep trees applied to the algorithm took a lot of time to execute however the RF model was much higher then DT. Considering the complexity of the data the research and hypotheses reported was lower; it took 1202.4895 seconds for DT to run and give predications much higher then we anticipated.
- For DT we obtained test error loss of 0.43581 and train error of 0.44182 .This we reduced significantly after optimizing the models.
- For Rf the predicted result came out was 0.029555 for the test error. In comparison to the paper [6] the writer used k-nearst neighbour classifier and found out test error was 2.4% . This is done on vector rather image and so this would be only useful for a baseline comparison. . Or Rf was used to compute a big amount of data as 60,00 pixels. Therefore in this study we managed to do the predication in about more then 1hrs.
- In comparison based on the result the two prediction was pretty good in terms of accuracy and also for RF we obtain 0 for training test error and so its important not to disregards the null hypotheses . From the result we can also consider that even though the accuracy for RF was higher at around 5% it still had expensive computation to DT. The reason behind this was the MINST dataset which had 60,00 training sets unlike if it was less dataset.
- To understand the closely the performance of the two algorithms we used confusion matrix to understand how well the classes are doing . When doing hyperparameter optimizing the RF took a lot of time around 25-30mins. From the graph the optimized model for DT some classes was not predicated therefore this might require to apply some smoothing techniques to solve this issues

LESSONS LEARNED AND FUTURE WORK

Lessons learned: RF performances are a high level in terms of predictive power and accuracy in contrast to DT. However the cons of this is significant factor of complexity Compute time and cost. With RF once can increase it's accuracy further and doing hyperparameter optimization can accelerate the performance of the algorithm.

Future work: To be able to use technique such as feature selection/engineering, compare the two methods using SMOTE, to use AUC – ROC curve to measure the performance , cross validation and additional manipulating hyperparameter optimization and adjusting other maximum and minimum features, leafs and splits.

Refernces

[1] N. Yann LeCun, Courant Institute, N. Y. Corinna Cortes, Google Labs, and R. Christopher J.C. Burges, Microsoft Research, 'THE MNIST DATABASE of handwritten digits'. [Online]. Available: http://yann.lecun.com/exdb/mnist/.

[2]S. Learns, '1.10. Decision Trees¶'. [Online]. Available: https://scikit-learn.org/stable/modules/tree.html.

[3]Abhishek Sharma, 'Decision Tree vs. Random Forest – Which Algorithm Should you Use?' [Online]. Available: https://www.analyticsvidhya.com/blog/2020/05/decision-tree-vs-random-forest-algorithm/.

[4]P. T. R, 'A Comparative Study on Decision Tree and Random Forest Using R Tool'.

[5]Kashish, 'KNN vs Decision Tree vs Random Forest for handwritten digit recognition'. [Online]. Available: https://medium.com/analytics-vidhya/knn-vs-decision-tree-vs-random-forest-for-handwritten-digit-recognition-470e864c75bc.

[6]N. Yann LeCun, Courant Institute, 'Comparision of leanring algorthims for hand written recogniation'.