# CSC730: Assignment 8
# How many labels do we really need, anyway?
# Active Learning

Kristophor Ray Jensen
*Electrical Engineering and Computer Science*
*South Dakota School of Mines and Technology*
Rapid City, United States
0009-0001-7344-349X

## I. Introduction

Anomaly detection is a crucial task in various domains, from fraud detection in financial transactions to identifying rare diseases in medical diagnosis [1]. Some times the data is not labeled, and it is expensive or difficult to label the data. In those instances active learning can be used to reduce the amount of labeled data required to train a model.

The requirements for this assignment are as follows [2].

1) Get the MNIST dataset (the original, balanced – not MNIST-C).
2) Write your own version of an active learning classifier. This should follow the basic outline shown in the figure above. Assume that the algorithm will start by querying 1 point at random. You may choose whatever utility function and classifier model you like. You do NOT need to write the actual classifier model itself from scratch.
3) Run your code on the dataset and determine accuracy and a confusion matrix.
4) Do this sequentially over a number of iterations sufficiently large to see performance flatten out, and plot accuracy vs. number of iterations. Show the confusion matrices at some selected points along the way.

Active learning is a machine learning paradigm that aims to reduce the amount of labeled data required to train a model [3]. In active learning, the model is allowed to query the user for labels of instances that it is uncertain about. This allows the model to focus on the most informative instances, reducing the need for large amounts of labeled data. In this assignment, we implement an active learning algorithm from scratch and evaluate its performance on the full MNIST dataset.

## II. Methodology

### A. Active Learning Algorithm Overview

The main components of the active learning algorithm include the following [3]:

- **Dataset:** The dataset consists of labeled and unlabeled instances. The labeled instances are used to train the model, while the unlabeled instances are used to select instances for labeling.
- **Learning Algorithm:** The classifier is a machine learning model that is trained on the labeled data. The classifier is used to make predictions on the unlabeled data and select instances for labeling.
- **Query Strategy:** The query strategy determines which instances to query for labels. The query strategy selects instances that the model is most uncertain about, based on the current model's predictions.
- **Expert:** The expert is the user who provides labels for the selected instances. The expert labels the instances selected by the query strategy, and the labels are used to update the model.

### B. Dataset

The MNIST data set consists of 70,000 labelled points. Each point is a 28x28 image of a digit from 0 to 9. The images are reshaped to a 1D array of length 784 and normalized to the range [0, 1]. The data set is balanced, with an equal number of images for each digit class.

In our example we created a 20% split of the data set to be used as the test set. The training dataset was then put into a pool of unlabeled data.

A second list of the training data was created to be used as the labeled data. This method reminds of of the psuedo-labeling method used in semi-supervised learning that we recreated in assignment 4.

Figure 1 shows a random selection of 16 nominal images from the training set.

### C. Learning Algorithm

There were two classifiers used in this assignment. The first classifier was a random forest classifier, and the second classifier was a support vector machine (SVM) classifier. The random forest classifier was used as the base classifier for the active learning algorithm. The SVM classifier was used to compare the performance of the active learning algorithm with a different classifier. The classifiers were trained on the

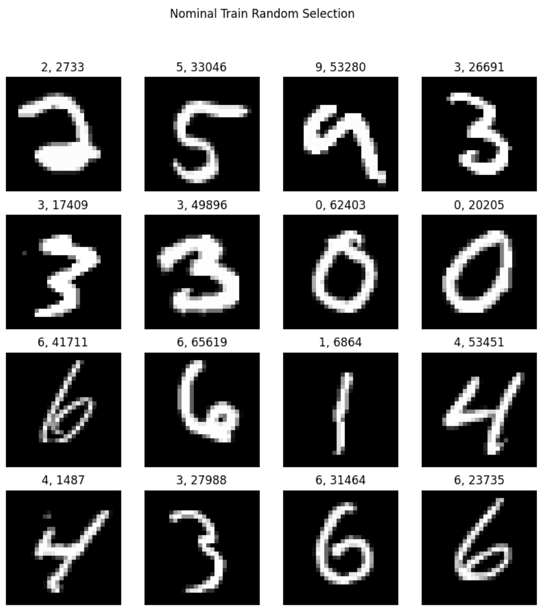

Figure 1. Random selection of nominal images from the training set

labeled data and used to make predictions on the unlabeled data. The predictions were used to select instances for labeling based on the query strategy. The classifiers were updated with the newly labeled instances, and the process was repeated for either a maximum number of iterations or minimum single-class accuracy reaching a threshold of 95%.

### *D. Query strategy*

In this assignment, we implemented the following query strategies:

- **Random Sampling:** Randomly select instances from the pool of unlabeled data for labeling. This strategy serves as a baseline for comparison with other query strategies.
- **Highest Confidence:** Select instances for labeling based on the classifier's highest confidence in its predictions. The highest confidence is calculated as the highest predicted probability for each instance.
- **Least Confidence:** Select instances for labeling based on the classifier's least confidence in its predictions. The least confidence is calculated as the difference between the highest and second-highest predicted probabilities for each instance.
- **Entropy Sampling:** Select instances for labeling based on the classifier's entropy of confidence in its predictions. The entropy is calculated as the negative sum of the predicted probabilities for each instance.
- **Query by Committee:** Select instances for labeling based on the disagreement among multiple classifiers. The committee consists of multiple classifiers trained on the labeled data, and the disagreement is calculated as the variance of the predicted probabilities for each instance.

### *E. Expert*

The expert in this assignment was simulated by selecting instances from the pool of unlabeled data for labeling. The instances were selected based on the query strategy, and the labels were used to update the model. The expert was assumed to provide accurate labels for the selected instances.

### *F. Evaluation Metrics*

Confusion matrices and accuracy scores were used to evaluate the performance of the active learning algorithm. The confusion matrix shows the number of true positives, false positives, true negatives, and false negatives for each class. The accuracy score is the proportion of correctly classified instances out of the total number of instances. The confusion matrix and accuracy score were calculated at each iteration of the active learning algorithm to track the performance of the model over time.

## III. Results and Discussion

There were a total of 10 trining simulations ran for this assigment. These simulations were broken up by two classifiers and five query strategies. The classifiers used were the SVM and the Random Forest. The query strategies used were Random, highest uncertainty, highest confidence, entropy, and query by committee. The results of these simulations are shown in the following tables.

| Classifier | Highest Uncertaintity | Highest Confidence | Random | Entropy | Query Committee |
|---|---|---|---|---|---|
| Rand Forest | 232 | 557 | 339 | 256 | 200 |
| SVM | 175 | 632 | 270 | 855 | 196 |

Table I
NUMBER OF QUERIES REQUIRED TO REACH 95% ACCURACY

The training was performed to a stopping condition of either 95% accuracy for the worst performing class or 1000 iterations. None of the simulations reached the 1000 iteration stopping condition. The SVM classifier performed better than the Random Forest classifier in most cases.

## IV. Conclusion

We were succesful in implementing an active learning algorithm to train a classifier on a dataset of 60000 samples using only a small set of those samples. The algorithm was able to reach a 95% accuracy in far less than 1000 iterations.

The SVM classifier performed better than the Random Forest classifier in most cases. The highest uncertaintity query strategy was the most effective query strategy for the SVM classifier. The query by committee strategy was the most effective query strategy for the Random Forest classifier. The random query strategy proved to be a quite effective query strategy for both classifiers. The entropy query strategy provided mixed results for each classifier.

Due to the size of the graphs, they are added to the appendix.

## References

[1] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422, 2008.

[2] R. Loveland, “Assignment_8.pdf,” From SDSMT D2L Website, 2024.

[3] A. Tharwat and W. Schenck, “A survey on active learning: State-of-the-art, practical challenges and research directions,” *Mathematics*, vol. 11, no. 4, p. 820, 2023. [Online]. Available: https://doi.org/10.3390/math11040820.

# APPENDIX

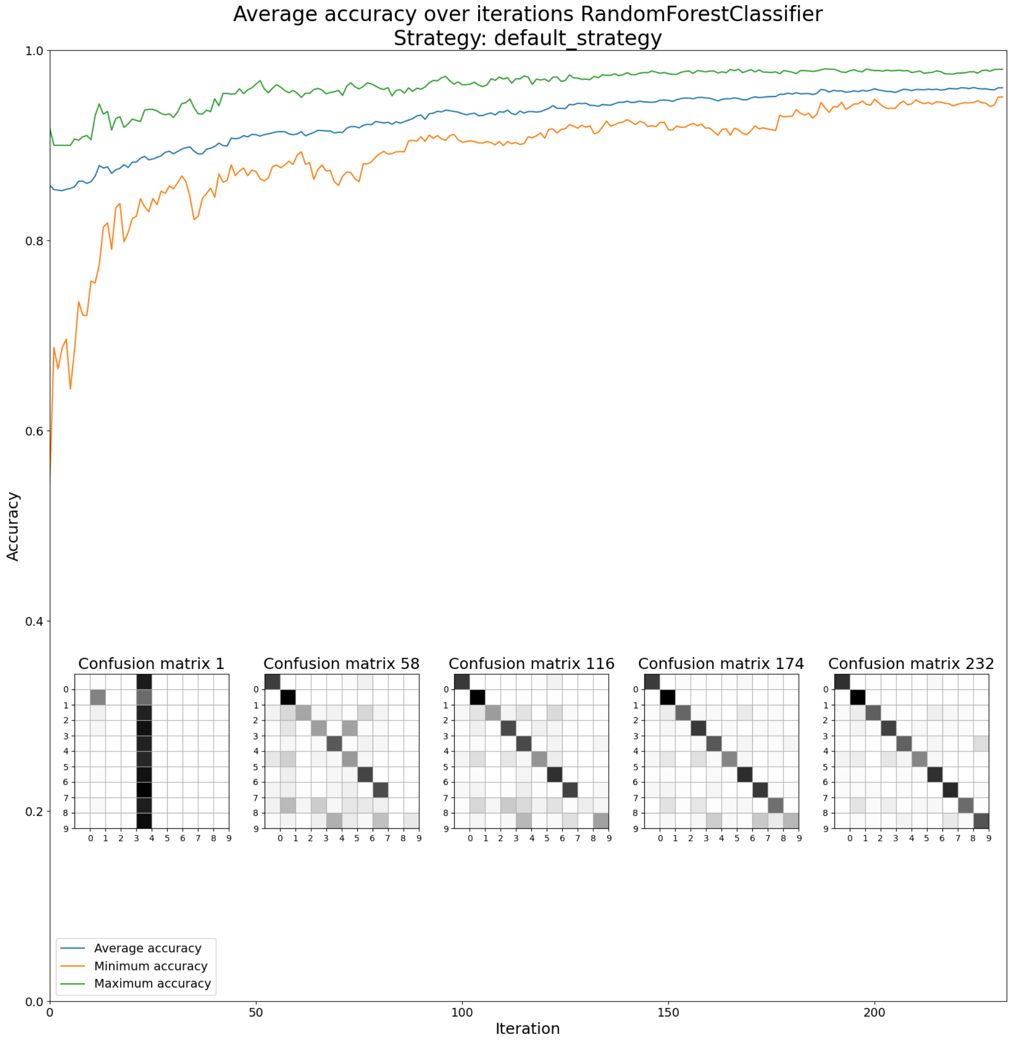

Figure 2. Random Forest Highest Uncertainty

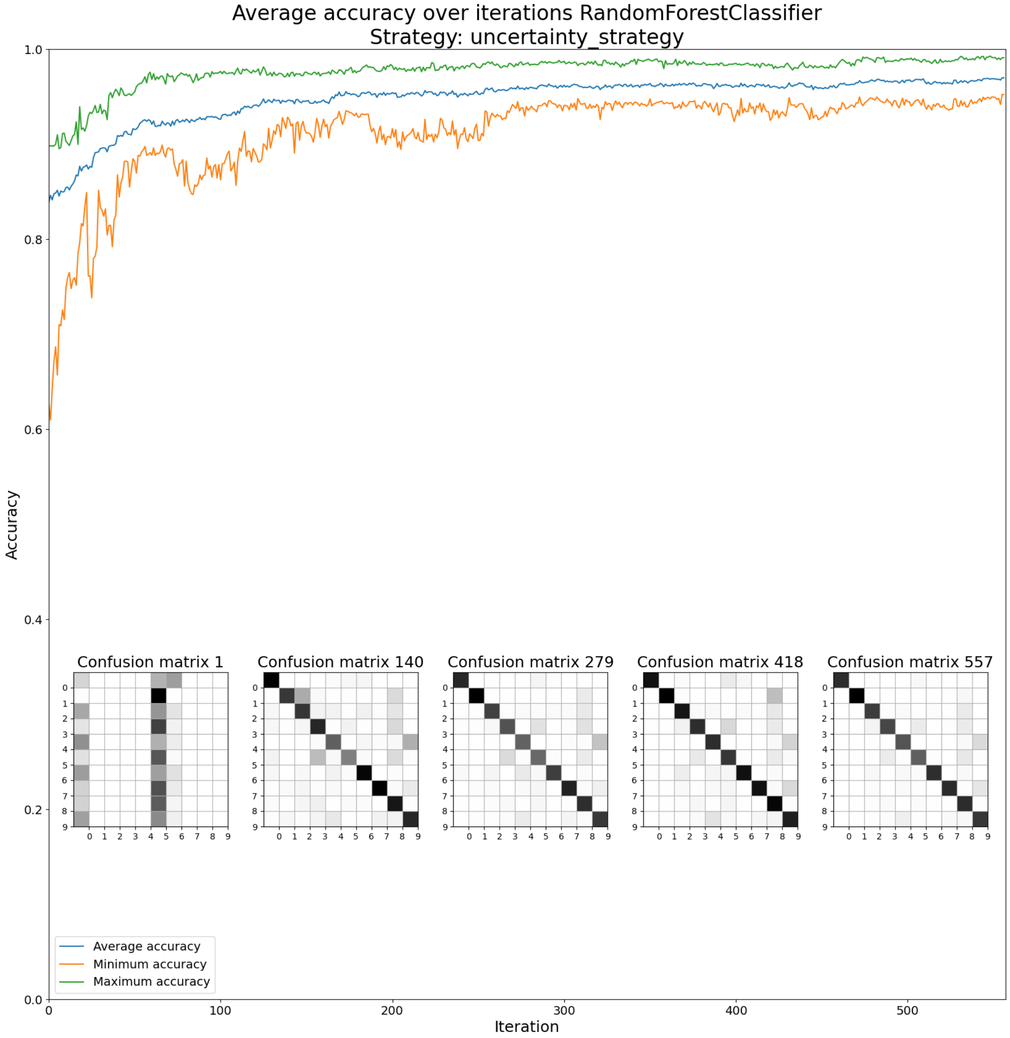

Figure 3. Random Forest Highest Confidence

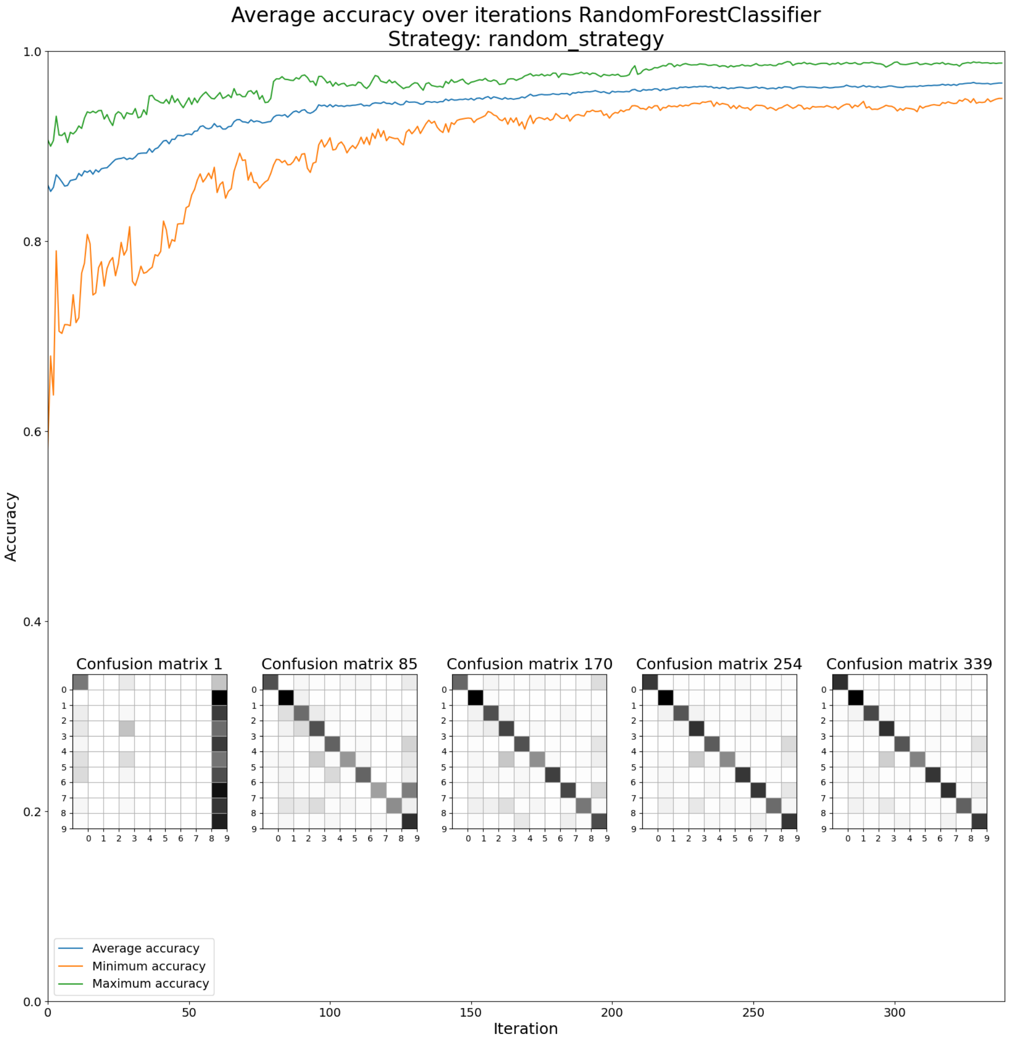

Figure 4. Random Forest Random

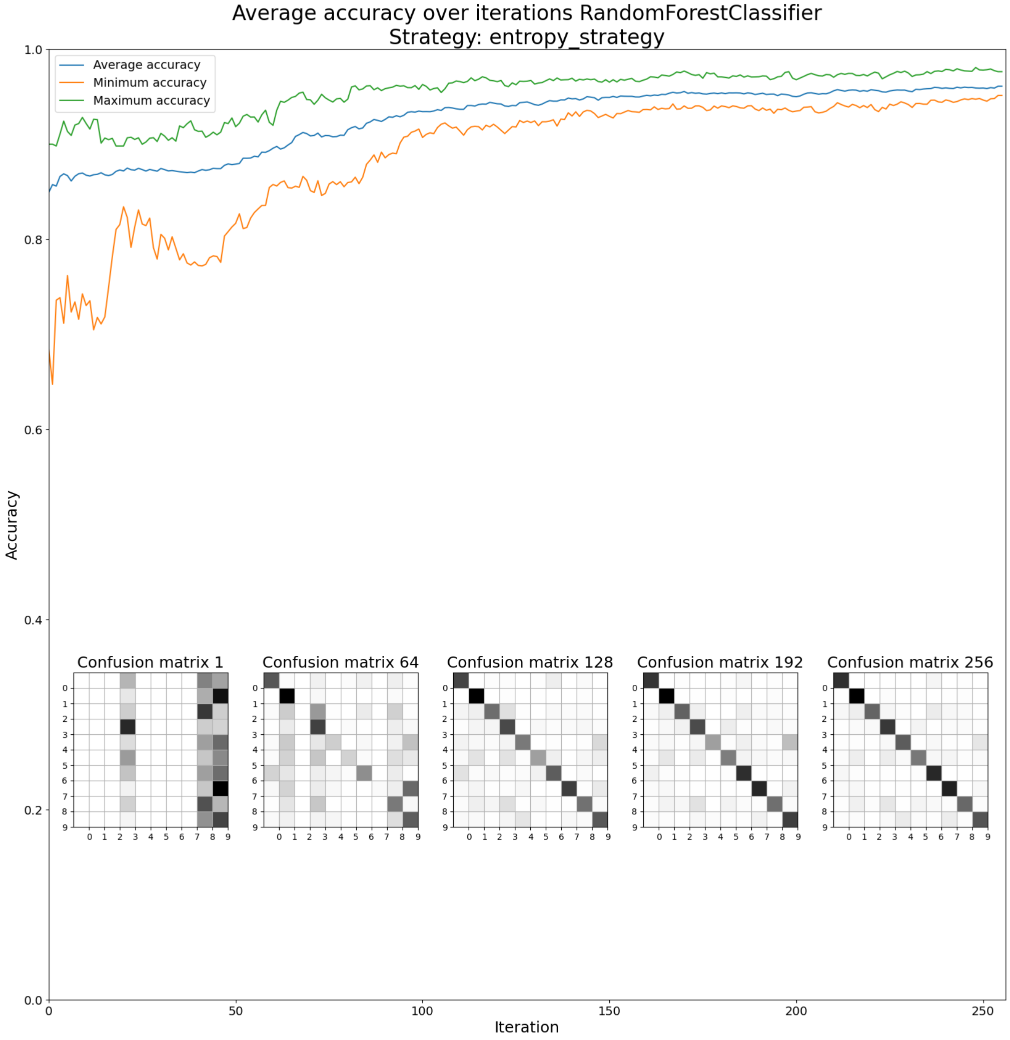

Figure 5. Random Forest Entropy

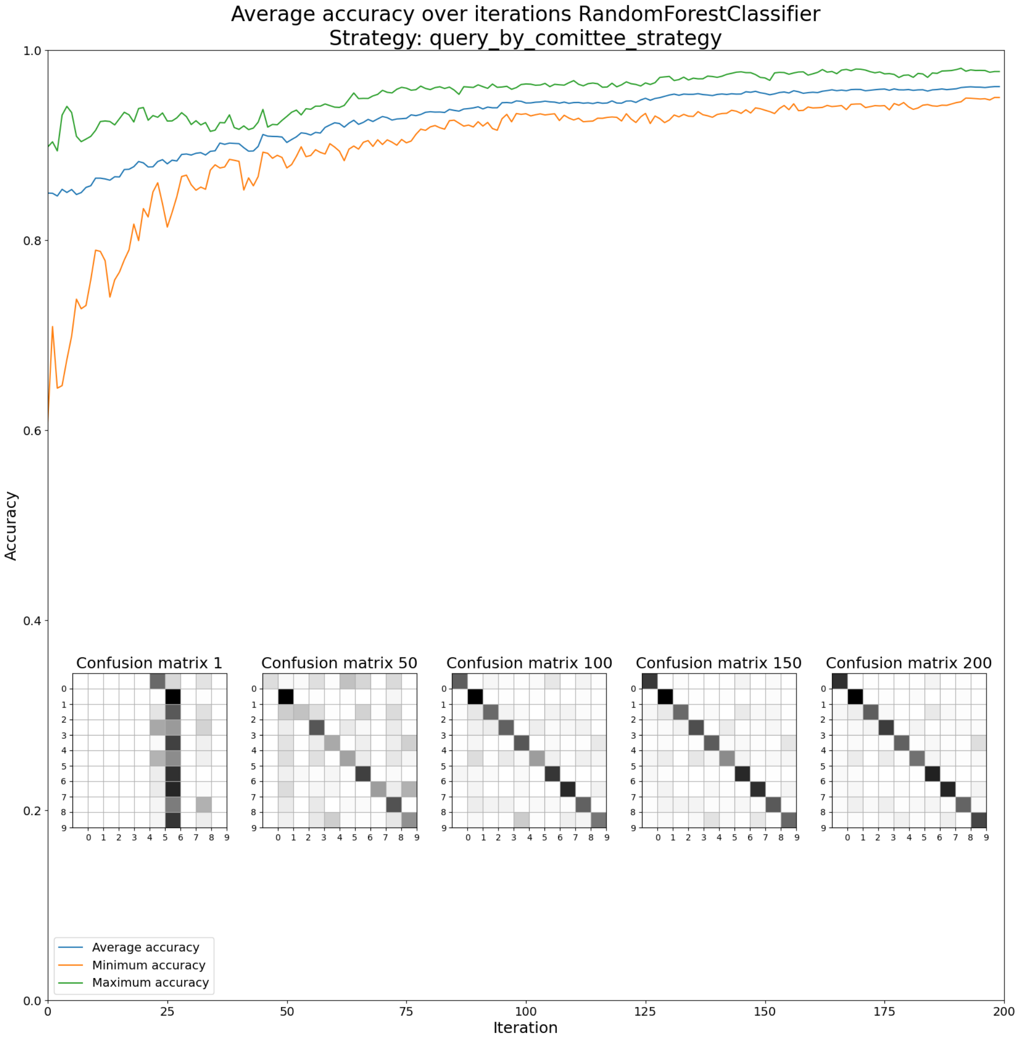

Figure 6. Random Forest Query by Committee

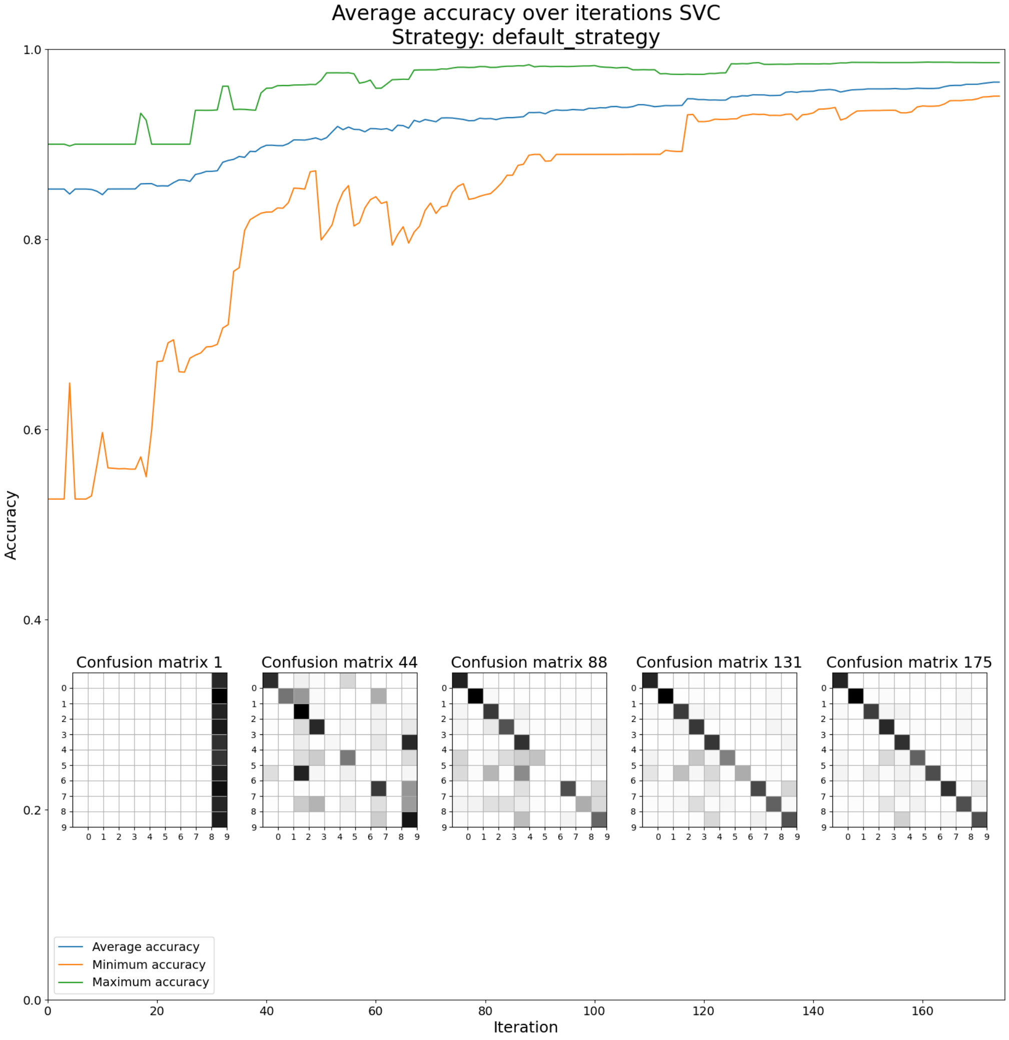

Figure 7. SVM Highest Uncertainty

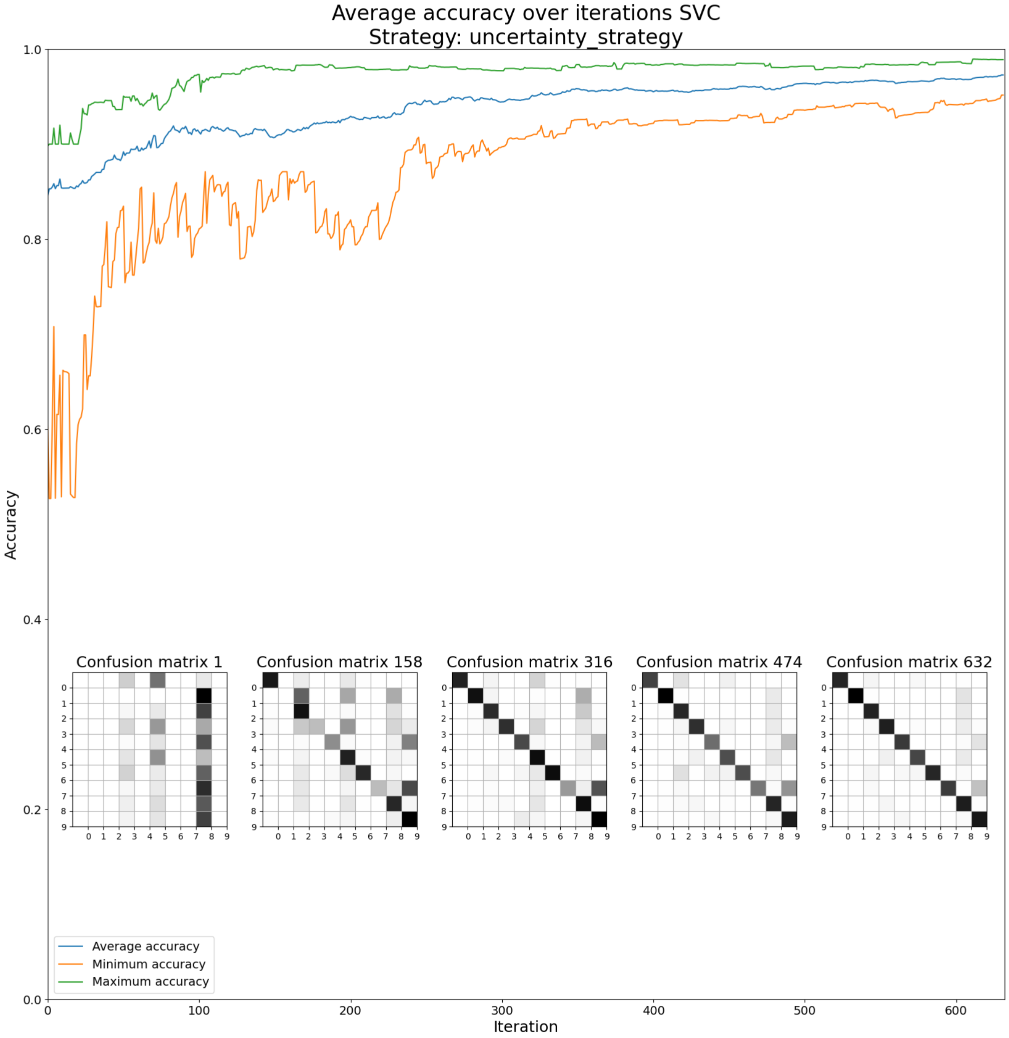

Figure 8. SVM Highest Confidence

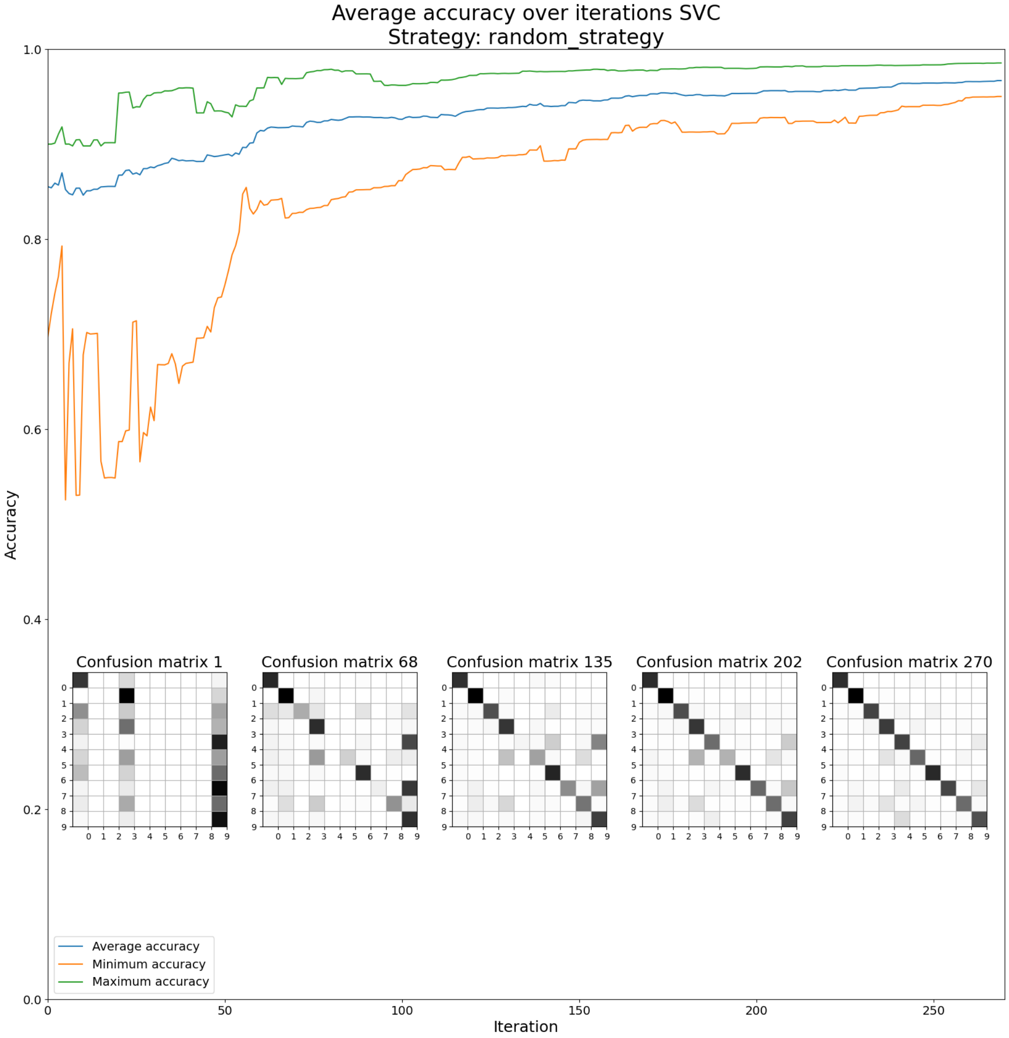

Figure 9. SVM Random

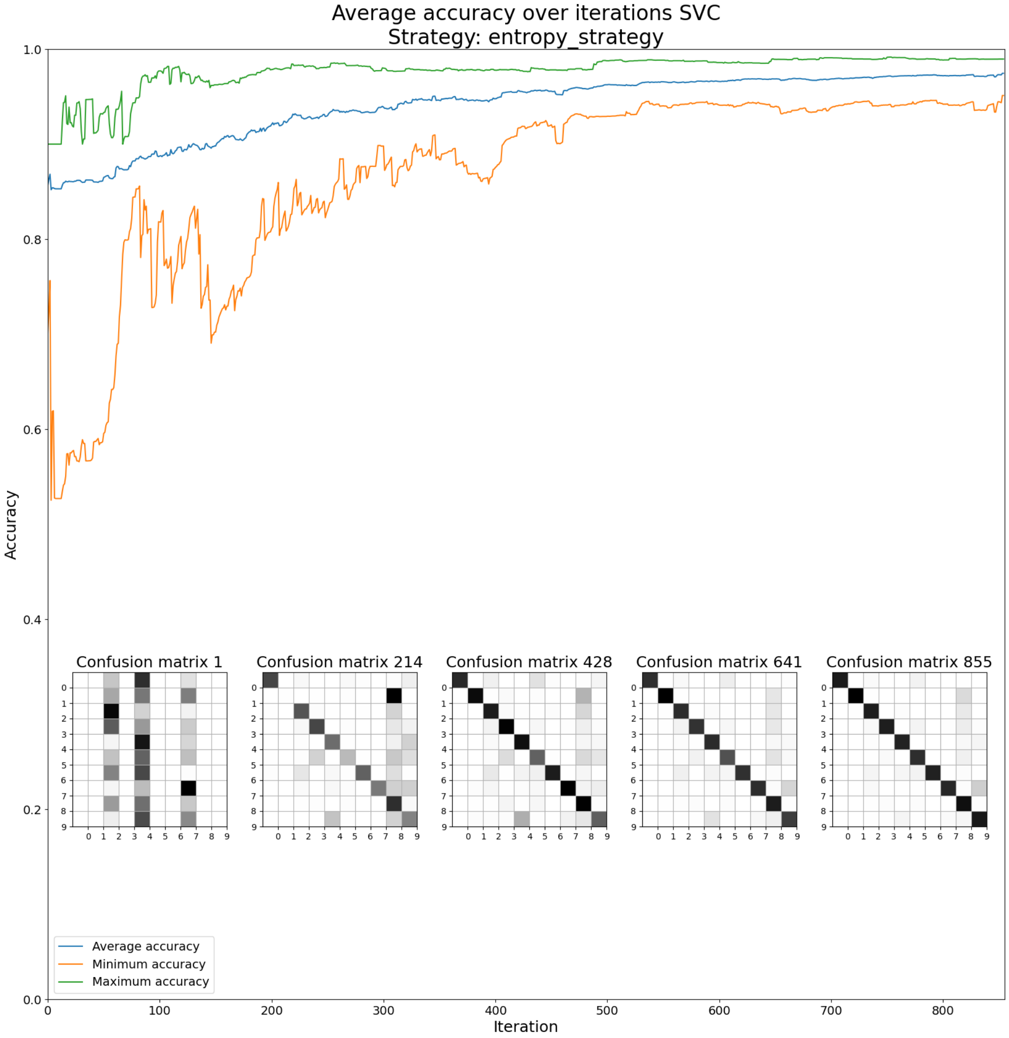

Figure 10. SVM Entropy

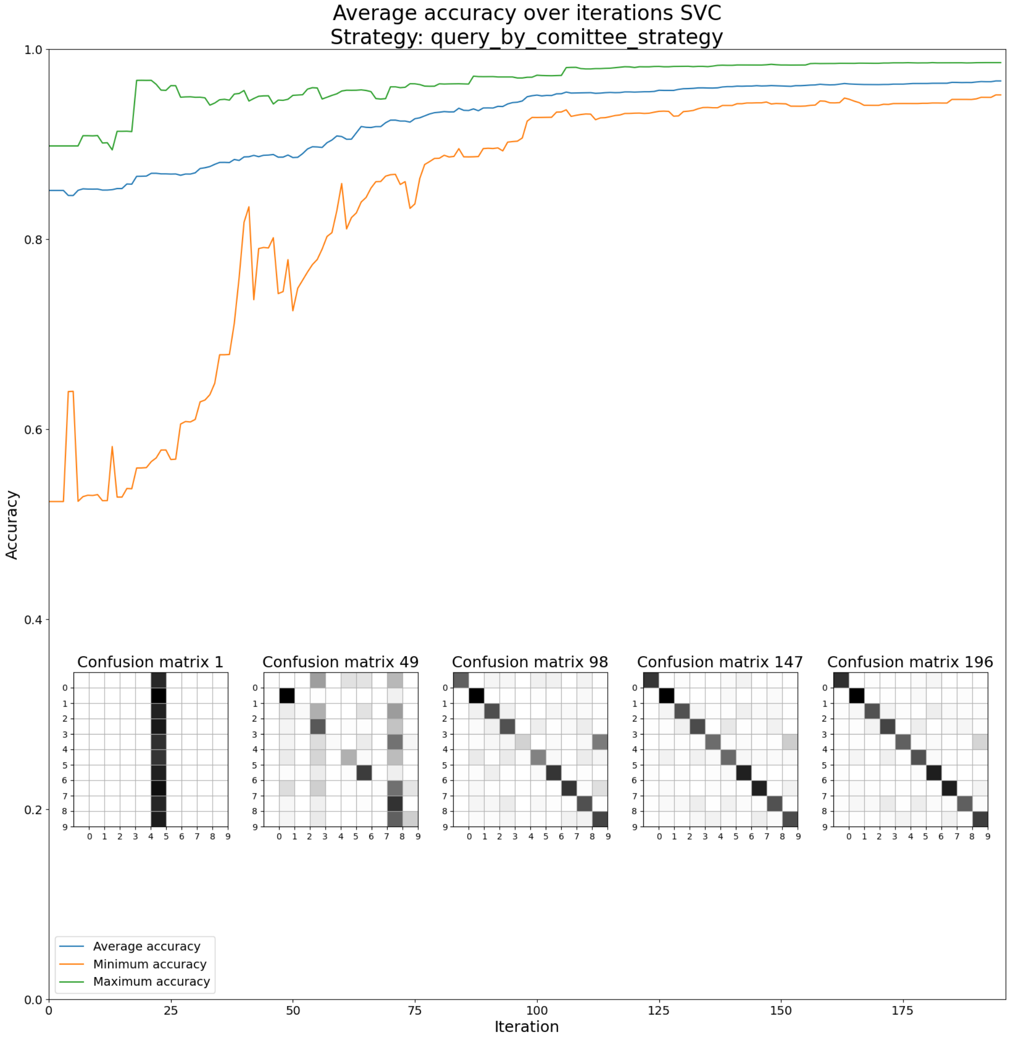

Figure 11. SVM Query by Committee