Assignment I
MNIST Handwritten Digits Classification

Adam Kosiorek

I. Objective

THE first assignment was to explore the SVM classifier, Decision Trees and Random Forests for the MNIST handwritten digits recognition.

II. Dataset

The MNIST dataset is comprised of 60000 training and 10000 testing samples [1]. Each sample is a 28x28 pixel black and white image of a single centered digit. It is widly used as a benchmark for comparing machine learning algorithms.

III. Experiments

All experiments were carried out on a notebook with Intel i7-2670QM quad core CPU and 8Gb RAM with Python SciPy package.

A. Dataset Preparation

It is known that processing input data prior to classification can improve results. In case of MNIST and SVM, however, state of the arts results have been achieved without preprocessing, that is, with raw pixels [2]. I examine the influence of mean intensity subtraction and momentum-based deskewing on classification accuracy.

B. Classifier Tuning

Most classifiers can be fine-tuned to a particular task by adjusting their hyperparameters. They are rarely independent of each other and it is necessary to perform an exhaustive grid search in the hyperparameter space. I chose to do so with 10% of the training data and 5-fold cross-validation. The parameters optimized for each classifier are shown in table I.

TABLE I
PARAMETERS OPTIMIZED FOR EACH CLASSIFIER

Classifier	Parameters
SVM	kernel type, C, gamma (for RBF), degree (for polynomial)
Decision Tree	cost function, max tree depth, number of features
Random Forests	number of estimators, cost function, max estimator depth, number of features

Classifiers with the best performing parameter combinations were trained on the full dataset.

IV. Results and Discussion

Table II summarizes results. Mean subtraction and deskewing applied together delieverd the highest accuracy for SVM. Therefore, this composition was used in the rest of the experiments. Parameter tuning resulted in the following parameters for each classifier:

- SVM: polynomial kernel of degree 2, $C = 1000$
- Decision Tree: 500 max features, 100 max depth, entropy criterion
- Random Forest: 50 max features, 10 max depth, entripy criterion, 100 estimators

TABLE II
RESULTS

Classifier	Preprocessing	Parameters	Accuracy
SVM	none	fine-tuned	98.02%
	mean		98.28%
	deskew		98.70%
	mean & deskew		98.84%
Decision Tree	mean & deskew	default	90.44%
	mean & deskew	fine-tuned	91.08%
Random Forest	mean & deskew	default	96.01%
	mean & deskew	fine-tuned	96.28%

In this setting SVM outperforms Decision Tree and Random Forest classifiers. Random Forest is far better than the basic Decision Tree, even though it uses the latter as its building block. Each tree of Random Forest's ensemble uses only up to 10% of the features and levels used by Decision Tree classifier.

Achieved performance is far from the state-of-the-art level, however. The highest classification performance up to date is by a convolutional neural network [3]. The best performing SVM classifier levereges robustness to affine-transformations by a special way of training [2].

Figures 1 and 2 depict pixel importance as seen by fine-tuned Decision Tree and Random Forest classifiers respectively. It seems that due to the limited depth and number of features the individual trees of Random Forest better cover the feature space of the MNSIT dataset, allowing for improved generalization.

References

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." Proceedings of the IEEE, 86(11):2278-2324, November 1998.

[2] D. Decoste and B. Schlkopf. "Training invariant support vector machines." Machine learning 46, no. 1-3 (2002): 161-190.

[3] Y. LeCun, C. Cortes, C.J.C. Burges. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.

[4] A. Ng, J, Ngiam, C.Y. Foo, Y. Mai and C Suen. "Unsupervised Feature Learning and Deep Learning Tutorial." http://ufldl.stanford.edu/wiki/index.php/UFLDL_Tutorial.

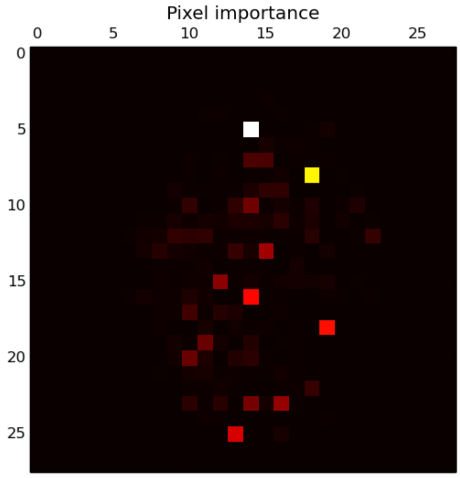

Fig. 1. Pixel importance for fine-tuned Decision Tree classifier

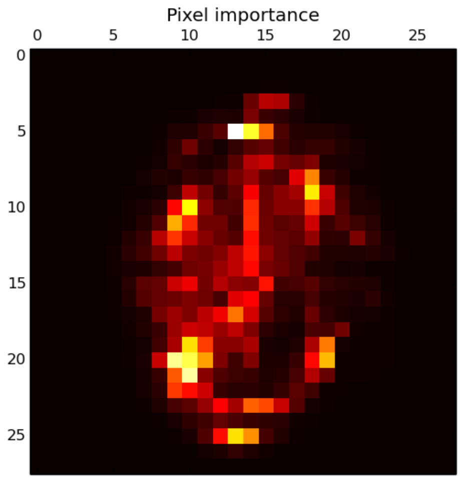

Fig. 2. Pixel importance for fine-tuned Random Forest classifier