Pattern Recognition Laboratory – Exercise #7
Recognition quality enhancement (handwritten digits)

Due date: **18.01.2018, 25.01.2018**

In the last exercise, your task will be to develop metaclassifier improving the quality of recognition of handwritten digits.

We'll be using MNIST database again. Using training data I prepared 7 different convolutional neural networks.

The procedure for the preparation of the classifiers was as follows:

1. MNIST training set was divided into two parts: the training part (50,000 samples) to be used by CNN training and the validation part (10,000 samples) left for metaclassifier preparation.

2. The 50k training set was randomly split in halves for training and validation, i.e. each convolutional network was trained on 25k samples and validated on 25k samples.

3. Validation set was classified with each of prepared classifiers. Files `valid_nb.txt` contain network output for all validation samples. (this means that we have 10 values for each sample). True labels of the samples in validation set are stored in `validlab.txt` file.

4. Also original test set from MNIST database was classified with all the classifiers. Files `test_nb.txt` contain network output for all test samples. True labels of the samples in test set are stored in `testlab.txt` file.

Recognition coefficients of elementary classifiers (in percent) contains the table below:

Classifier	#1	#2	#3	#4	#5	#6	#7
Recognition quality on validation set	98,30	98,30	98,56	98,49	98,60	98,50	98,34
Recognition quality on test set	98,58	98,58	98,54	98,63	98,69	98,71	98,49

Reference solution uses the simplest method of making decisions based on the output values (highest value) and standard methods of voting. The first requires the unanimity of all classifiers; second absolute majority of votes (with the seven classifiers this gives 4 votes); in the third label is chosen with the highest number of votes, unless another digit does not get the same number of votes.

The results are presented in the following table:

	Unanimity voting			Absolute majority			Majority voting		
	OK.	Error	*Reject*	OK.	Error	*Reject*	OK.	Error	*Reject*
Coefficients and objective function for validation set	97,29	0,68	2,03	98,61	1,36	0,03	98,61	1,36	0,03
	90,49			85,01			85,01		
Coefficients and objective function for test set	97,59	0,55	1,86	98,66	1,22	0,12	98,68	1,25	0,07
	92,09			86,46			86,18		

To evaluate the solution the following objective function is used:

```
fobj = recognition_coefficient – beta * error_coefficient
```

where `beta` is the ratio of error decision cost to the cost of reject decision (in our case `beta = 10`).

I expect you to develop different method of improving the quality, than used in the reference solution. Obtaining maximum points from this exercise requires the objective function value higher than the reference solution.

The important thing, which you need to pay attention to is the ***wear*** of the test set. In preparing the reference solution I didn't not used at all provided results of classification of the test. Will you compute any parameters (e.g. weights of classifiers) they should be derived only from the validation set (possibly divided into training and validation parts). In the latter case, a reasonable approach would be drawing a half of the samples to the training set and using the remaining samples as a test set. This procedure of random partitioning (training and quality measurement) should be repeated at least 3 times to make sure that the results are repeatable - in other words, you developed a reasonable metaclassifier.

You should report (in the laboratory oral report is OK ☺):

1. Description of the method of assembling the results of the individual classifiers, with particular regard to the parameters computed in the metaclassifier's training.

2. Metaclassifier's training algorithm description (if any).

3. Data on the quality of recognition of digits and values of the objective function.

Files attached to this instruction:

valid_*.txt   outputs of individual networks for validation set

test_*.txt   outputs of individual networks for test set

validlab.txt   validation samples labels

testlab.txt   test samples labels

*.m   reference solution implementation