Pattern Recognition: Exercise 2a

The goal of this exercise was to use the SVM on the `MNIST` dataset[1]. It is separated in two main parts. The first one is to optimize the hyperparameters and the second one is to train and then test the model with the best hyperparameters on the dataset.

Hyperparameters optimization

The goal of this first part is to get quickly an idea of which hyperparameters we need to use in order to train the model on the dataset. To do that, we first choose two kernels: **linear** and **radial basis function** (RBF). For the linear kernel, we need to optimize only one parameter, that is C. It gives us a trade-off between choosing to maximize the margin, or minimizing the misclassified distances. For the RBF kernel, we need to optimize two parameters, that are the C parameter as before and also the γ parameter, that is the inverse of the standard deviation, i.e. a small γ will treat two points as similar even if they are far from each other.

Values of both parameters are tested in a logarithmic range of basis 10, with exponent -5 to 5 for C, and with exponent -10 to 1 for γ. These values are combined with each other, and they are trained and validated on a subset of the dataset containing 2000 random samples (about 4% of the total dataset) using a **5-cross validation**. The average accuracy is retained after the 5-cross validation, and the hyperparameters that gave the best accuracy are kept for the second part. The results are shown in a table in the next pages.

For our run, the best parameters are: **RBF kernel with** $C = 10$ **and** $\gamma = 10^{-7}$ with an accuracy of 0.9305. Note that we used a seed for reproducibility of the experiment. 2000 samples out of 50000 are enough to get an idea of the best parameters, because most of the runs have a very poor accuracy ($< 20\%$). As we look at the results, for the RBF kernel it seems that γ has very much more impact on the accuracy than the C parameter. The only condition seems to be that $C \geq 1$ to be efficient. Also, for the linear kernel, the C parameter has not much impact on the accuracy, or we may need more data to have a significant difference in the accuracy for this type of kernel.

Training

After optimizing the hyperparameters, we train the full dataset with theses hyperparameters. The training part is short (about 2,5 minutes for the full dataset). After the training, we will evaluate the model on the test set, and compute the accuracy. With our optimized parameters we get an accuracy of **0.9823**.

[1] `http://yann.lecun.com/exdb/mnist/`

Results of the hyperparameters optimization

The results are sorted by accuracy. The hyperparameters in red on the first line are used in the training part.

Kernel	C	Gamma	Accuracy	Runtime (seconds)
rbf	10.0	1e-07	0.9305	1.576
rbf	10000.0	1e-07	0.93	1.589
rbf	1000.0	1e-07	0.93	1.623
rbf	100.0	1e-07	0.93	1.597
rbf	100000.0	1e-07	0.93	1.602
rbf	1.0	1e-07	0.9125	1.955
rbf	100000.0	1e-06	0.905	4.432
rbf	10.0	1e-06	0.905	4.334
rbf	1000.0	1e-06	0.905	4.433
rbf	100.0	1e-06	0.905	4.336
rbf	10000.0	1e-06	0.905	4.431
rbf	1.0	1e-06	0.902	4.397
rbf	10000.0	1e-08	0.9	1.319
rbf	100000.0	1e-08	0.9	1.332
rbf	1000.0	1e-08	0.9	1.3
rbf	100.0	1e-08	0.8995	1.318
rbf	10.0	1e-08	0.8975	1.591
rbf	100000.0	1e-09	0.894	1.28
rbf	1000.0	1e-10	0.894	1.605
rbf	10000.0	1e-09	0.894	1.291
rbf	100.0	1e-09	0.894	1.58
rbf	1000.0	1e-09	0.8935	1.279
rbf	10000.0	1e-10	0.892	1.296
linear	1e-05		0.892	1.02
linear	0.0001		0.892	0.993
linear	1000.0		0.892	1.049
linear	0.001		0.892	0.992
linear	0.01		0.892	0.987
linear	0.1		0.892	0.99
linear	1.0		0.892	0.987
linear	10.0		0.892	0.991
linear	100.0		0.892	1.023
linear	100000.0		0.892	1.038
linear	10000.0		0.892	0.999
rbf	100000.0	1e-10	0.8915	1.283
rbf	100.0	1e-10	0.8345	3.132
rbf	10.0	1e-09	0.834	3.104
rbf	1.0	1e-08	0.8325	3.159
rbf	0.1	1e-07	0.808	3.534
rbf	0.1	1e-06	0.213	4.384
rbf	10.0	1e-10	0.169	4.345
rbf	1.0	1e-09	0.167	4.245
rbf	0.1	1e-08	0.1525	4.343
rbf	100.0	1e-05	0.1305	4.601
rbf	10.0	1e-05	0.1305	4.712
rbf	1000.0	1e-05	0.1305	4.605
rbf	100000.0	1e-05	0.1305	4.663
rbf	10000.0	1e-05	0.1305	4.666
rbf	1.0	1e-05	0.1295	4.677

Kernel	C	Gamma	Accuracy	Runtime (seconds)
rbf	100000.0	1.0	0.1095	4.67
rbf	100000.0	0.1	0.1095	4.712
rbf	100000.0	0.01	0.1095	4.671
rbf	100000.0	0.001	0.1095	4.719
rbf	100000.0	0.0001	0.1095	5.16
rbf	10.0	0.01	0.1095	4.766
rbf	10000.0	10.0	0.1095	4.686
rbf	10000.0	1.0	0.1095	4.67
rbf	10000.0	0.1	0.1095	4.685
rbf	10000.0	0.01	0.1095	4.672
rbf	10000.0	0.001	0.1095	4.692
rbf	10.0	0.0001	0.1095	5.246
rbf	10.0	0.001	0.1095	4.737
rbf	100.0	10.0	0.1095	4.628
rbf	10000.0	0.0001	0.1095	5.154
rbf	100.0	1.0	0.1095	4.644
rbf	1000.0	10.0	0.1095	4.624
rbf	10.0	0.1	0.1095	4.759
rbf	10.0	1.0	0.1095	4.766
rbf	10.0	10.0	0.1095	4.757
rbf	1000.0	1.0	0.1095	4.635
rbf	1000.0	0.1	0.1095	4.622
rbf	1000.0	0.01	0.1095	4.637
rbf	1000.0	0.001	0.1095	4.632
rbf	1000.0	0.0001	0.1095	5.122
rbf	1.0	1.0	0.1095	4.673
rbf	100.0	0.0001	0.1095	5.13
rbf	100.0	0.001	0.1095	4.659
rbf	100.0	0.01	0.1095	4.665
rbf	100.0	0.1	0.1095	4.632
rbf	1.0	10.0	0.1095	4.617
rbf	1.0	1e-10	0.1095	4.26
rbf	1.0	0.1	0.1095	4.656
rbf	0.0001	1e-08	0.1095	4.741
rbf	0.001	1e-07	0.1095	4.748
rbf	0.001	1e-08	0.1095	4.723
rbf	0.001	1e-09	0.1095	4.753
rbf	0.001	1e-10	0.1095	4.7
rbf	0.0001	10.0	0.1095	4.735
rbf	0.0001	1.0	0.1095	4.827
rbf	0.0001	0.1	0.1095	4.839
rbf	0.0001	0.01	0.1095	4.758
rbf	0.0001	0.001	0.1095	4.807
rbf	0.0001	0.0001	0.1095	5.217
rbf	0.0001	1e-05	0.1095	4.726
rbf	0.0001	1e-06	0.1095	4.808
rbf	0.0001	1e-07	0.1095	4.735
rbf	0.0001	1e-09	0.1095	4.76
rbf	1.0	0.01	0.1095	4.668
rbf	0.0001	1e-10	0.1095	4.798
rbf	1e-05	10.0	0.1095	4.83
rbf	1e-05	1.0	0.1095	4.807
rbf	1e-05	0.1	0.1095	4.792
rbf	1e-05	0.01	0.1095	4.884

Kernel	C	Gamma	Accuracy	Runtime (seconds)
rbf	1e-05	0.001	0.1095	4.81
rbf	1e-05	0.0001	0.1095	5.233
rbf	1e-05	1e-05	0.1095	4.772
rbf	1e-05	1e-06	0.1095	4.989
rbf	1e-05	1e-07	0.1095	4.903
rbf	1e-05	1e-08	0.1095	5.03
rbf	1e-05	1e-09	0.1095	4.906
rbf	1e-05	1e-10	0.1095	4.984
rbf	0.001	1e-06	0.1095	4.839
rbf	0.001	1e-05	0.1095	4.738
rbf	0.001	0.0001	0.1095	5.391
rbf	0.001	0.001	0.1095	5.021
rbf	1.0	0.001	0.1095	4.663
rbf	1.0	0.0001	0.1095	5.147
rbf	0.1	10.0	0.1095	4.531
rbf	0.1	1.0	0.1095	4.54
rbf	0.1	0.1	0.1095	4.523
rbf	0.1	0.01	0.1095	4.526
rbf	0.1	0.001	0.1095	4.521
rbf	0.1	0.0001	0.1095	5.036
rbf	0.1	1e-05	0.1095	4.582
rbf	0.1	1e-09	0.1095	4.356
rbf	0.1	1e-10	0.1095	4.374
rbf	0.01	10.0	0.1095	4.618
rbf	0.01	1.0	0.1095	4.793
rbf	0.01	0.1	0.1095	5.077
rbf	0.01	0.01	0.1095	5.147
rbf	0.01	0.001	0.1095	5.041
rbf	0.01	0.0001	0.1095	5.466
rbf	0.01	1e-05	0.1095	5.065
rbf	0.01	1e-06	0.1095	4.858
rbf	0.01	1e-07	0.1095	4.817
rbf	0.01	1e-08	0.1095	4.777
rbf	0.01	1e-09	0.1095	4.76
rbf	0.01	1e-10	0.1095	4.862
rbf	0.001	10.0	0.1095	5.057
rbf	0.001	1.0	0.1095	5.042
rbf	0.001	0.1	0.1095	4.978
rbf	0.001	0.01	0.1095	5.021
rbf	100000.0	10.0	0.1095	4.714