Machine Learning, Spring 2018: Exercise Sheet 8

This is a programming exercise. It will be graded and the grade counts toward the course grade. ***Join into groups of two or three*** *and submit a single solution per group, indicating the group members' names on the report sheet. You can use Python or Matlab.*

If you cannot find a team to join until April 6, send me an email. I will then swiftly create teams of 2 from the team-seekers by coupling them in alphabetical order. After that I will not assist creating teams. Homeworks submitted by "singletons" will be corrected but not counted toward the course grade.

Please email your solutions to our two TA's Tianlin Liu (t.liu@jacobs-university.de) and Tayyab Mateen (t.mateen@jacobs-university.de)

Deadline for submission is Sunday April 15, 23:59 (email timestamp). Submissions arriving later (even a second after midnight) will be corrected but not counted for the course grade. Ask me in class why I apply such a strict deadline rule and I will tell you a real-life story.

Summary. The objective of this task is to train a suite of classifiers of increasing model flexibility along the outlines summarized in the LN Section, and document the under/overfitting phenomenon in a graphic analog to Figure 20 in the LN.

Detailed task description. The objective is to train a number of different classifiers for the digits data that you still can find at

http://minds.jacobs-university.de/sites/default/files/uploads/teaching/share/DigitsBasicRoutines.zip .

I will describe how to get a suite of *different* classifiers further below and start with describing how to train a single classifier.

The classification task is to classify the 10 digit classes, using the original benchmark set-up. That is, from each of the 200 examples given in the dataset for each digit, take the first 100 as *training data* and the second 100 as *testing data*. Throughout the training process, never touch the testing data – pretend they don't exist. Only *after* you have a trained a classifier solely using the training data, you may test it on the testing data.

Here are the steps you must carry out in order to train one classifier:

1. Decide on a set of k features that you use to trim the input pattern dimension from 240 to k. The choice of features is up to you (hand-made, k-means based, or PCA features). Apply these features, obtaining k-dimensional feature vectors $\mathbf{f}^k(x_i)$, one such feature vector for each of your 10×100 training patterns x_i.
2. Let us number the classes as class 1 = the "1" patterns, …, class 9 = the "9" patterns, class 10 = the "0" patterns. For each training pattern x_i create a binary, 10-dimensional *target vector* z_i , which is zero everywhere except at the position corresponding to the class number of the pattern x_i, where it is set to 1.

3. Compute the linear regression weights W_{opt} to approximate z_i from $\mathbf{f}^k(x_i)$, (padded with a trailing 1 as bias, so $\mathbf{f}^k(x_i)$ actually has size $k + 1$). W_{opt} should be a matrix of size $10 \times (k + 1)$. Use the formula (25) from the LNs – do not use ridge regression. While ridge regression is what a professional expert would use, we do not use it in this HW because it may conceal the effects of overfitting which I want you to experience.
4. For each pattern x_i, compute the square *training error* $\varepsilon_{i,train}^2 = \left\| W_{opt}\mathbf{f}^k(x_i) - z_i \right\|^2$ and average the 1,000 such errors, getting the *mean square training error* $\text{MSE}^k_{\text{train}}$. Furthermore, compute the *training misclassification rate* $\text{MISS}^k_{\text{train}}$, that is the ratio of misclassified patterns over the total number of patterns (= 1000). A pattern is misclassified if the index of the maximum value in $W_{opt}\mathbf{f}^k(x_i)$ is not the number of the correct pattern.
5. Now consider the 1,000 patterns from the testing data. Call them t_j. For each of them compute the square *testing error* $\varepsilon_{j,test}^2 = \left\| W_{opt}\mathbf{f}^k(t_j) - z_j \right\|^2$ and their average $\text{MSE}^k_{\text{test}}$. Also compute $\text{MISS}^k_{\text{test}}$.
6. The outcome such a training trial is thus comprised of the four numbers $\text{MSE}^k_{\text{train}}$, $\text{MISS}^k_{\text{train}}$, $\text{MSE}^k_{\text{test}}$, $\text{MISS}^k_{\text{test}}$.

Now I explain how to get a sequence of different classifiers. You get the first classifier by using only a single feature (k = 1!) – besides the bias feature which always is tacitly included. Carry out the above 6-step procedure and get $\text{MSE}^1_{\text{train}}$, $\text{MISS}^1_{\text{train}}$, $\text{MSE}^1_{\text{test}}$, $\text{MISS}^1_{\text{test}}$. Next, add one feature (do not change the first feature), getting $\text{MSE}^2_{\text{train}}$, $\text{MISS}^2_{\text{train}}$, $\text{MSE}^2_{\text{test}}$, $\text{MISS}^2_{\text{test}}$. Continue this game until $k_{\text{max}} = 800$ or until you run out of features, whichever comes first (try to scale up to at least 200 features). Then plot the sequences $(\text{MSE}^k_{\text{train}})_{k=1,\ldots,kmax}$, $(\text{MSE}^k_{\text{test}})_{k=1,\ldots,kmax}$, $(\text{MISS}^k_{\text{train}})_{k=1,\ldots,kmax}$, $(\text{MISS}^k_{\text{test}})_{k=1,\ldots,kmax}$, obtaining a plot like Figure 20 in the LN. Create 2 plots, each of them showing all 4 sequences, one of the plots using logarithmic scaling and the other linear scaling.

Deliverables: 1. A typeset report, target size 1 page excluding figure(s), in which you specify which features you used. For a very good grade, besides the technical documentation of your work, an insightful discussion of the results and/or extensions of the basic experiment are expected (for instance, try and discuss different kinds of features). 2. Your code, which should be documented inline well enough to allow the TAs to quickly orient themselves within the code.

The grade will be based solely on the report, except when the TAs need to inspect the code in order to understand your report or to give helpful hints and then find that your code under-documented, which will lead to a grade reduction.