T-61.3050 Data Challenge Submission

Olli Jarva & Jarno Rantanen

Aalto University School of Science

olli@jarva.fi & jarno@jrw.fi

1 Solution in a nutshell

We're using Support Vector Machines (SVM's) for classifying the input data (instances of hand-written characters) into 26 classes corresponding to the english alphabet. Images are preprocessed by aligning them to the lower left corner. The SVM uses a Radial Basis Function (RBF) kernel, with chained second-level SVM's specialized in entries commonly misclassified in the training data. The optimal kernel function parameters are sought for using standard grid-search and validated using k-fold cross-validation. With k=20, we achieved 89.6% classification performance over training set.

2 Detailed description

SVM's were chosen for their relative ease of use, while still being a robust and well-performing machine learning method for high-dimensional data. Implementing an SVM from scratch is relatively error-prone, so we're using the open-source `mlpy` Python library for a ready-made SVM implementation based on `libsvm`.

The RBF kernel function was chosen simply because our literature review suggested them as a reasonable default choice with SVM. Also having only two parameters to the SVM helps in its training, as the search space is only two-dimensional (as opposed to the 4 dimensions of the polynomial kernel, for example).

The training data uncovered sets of characters commonly misclassified by the primary classifier. While the characters themselves weren't easily classified, it was fairly easy to confidently identify the *sets*. For each such set, the classification was then delegated to a second-level SVM, specifically trained against that subset of the original problem. With the latest training runs, we found this to increase our classification performance by almost 0.4%, which we considered quite satisfactory. Together the SVM's form a *classifier tree* of height 2. Adding third level to the *classifier tree* made code more complex, and we couldn't distinguish possible advantage from measurement errors.

Optimal parameters for the SVM (and the kernel function) present a maximization problem in a 2-dimensional search space. Since an exhaustive search in this space isn't possible, a basic grid search was employed. The performance of the classifier tree in each grid point was evaluated using k-fold cross-validation. Cross-validation was chosen as it provides reasonable protection against overfitting against the training data.