1. Computer Lab
(Statistical Machine Learning)

Aleš Kučera

7. prosince 2023

In this computer lab, our objective was to identify a name given an image of handwritten characters.

Tasks 1.-3.

The initial three classifiers were implemented according to the assigned tasks. Code details are omitted here as they mirror the task requirements.

Task 4.

Table 1 displays the error rates of the implemented classifiers from tasks 1.-3.

Classifier	R^{seq} [%]	R^{char} [%]
Independent multi-class classifier (task 1)	70.8	26.2
Structured pair-wise dependency (task 2)	13	5.76
Structured fixed number of sequences (task 3)	1.8	0.47

Tabulka 1: Testing errors of the first three classifiers.

The optimal performance is observed with the third perceptron. This is attributed to leveraging comprehensive prior information about the words. This approach is particularly advantageous as it involves breaking down the classification problem into smaller, more manageable components. By focusing on the classification of words of a specific length, the potential word candidates are substantially reduced.

The second perceptron also exhibits commendable results, albeit not reaching the level of the third one. This is largely due to incorporating information about the sequence of characters, which contributes to its solid performance.

Conversely, the first perceptron demonstrates the least favorable outcome. This outcome is anticipated, considering that predictions are solely made on a character-by-character basis. Consequently, the prediction of the next character is unaffected by the preceding characters, presenting a significant drawback given the interdependence of characters within a word.

Task 5.

Task 5 involved implementing the Structured Output SVM. In comparison to task 2, we did not require the classifier to be only correct, but we enforced correctness above the margin, specifically the Hamming distance. Additionally, regularization loss was introduced in the weights update, and the update of weights was weighted by a decaying learning rate.

We elaborate on the changes made:

1. **Addition of the Hamming distance:** The key modification involves adjusting the values of q as follows:

$$q_i(y_i) = \langle \mathbf{w}_{y_i}, \mathbf{x}_i \rangle + b_i + l(y_i, y) \tag{1}$$

where y is the true label, and $l(a, b)$ represents the Hamming distance between words a and b.

2. **Addition of regularization:** The update for both weights $\mathbf{W}$ and function g weights $\mathbf{G}$ is as follows:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \alpha_k \cdot \lambda \cdot \mathbf{W}_k \tag{2}$$

$$\mathbf{G}_{k+1} = \mathbf{G}_k - \alpha_k \cdot \lambda \cdot \mathbf{G}_k \tag{3}$$

$$\tag{4}$$

   where λ is the regularization strength, and α_k is the learning rate for iteration k.

3. **Change in the update of weights:** The final change involves multiplying the update when the prediction is incorrect by α_k, where:

$$\alpha_k = \frac{\alpha}{k} \tag{5}$$

Table 2 presents a comparison of all classifiers. Notably, the results for the last task are the same as those for task 2, but this is coincidental. Adjusting hyperparameters could yield different outcomes.

Classifier	R^{seq} [%]	R^{char} [%]
Independent multi-class classifier (task 1)	70.8	26.2
Structured pair-wise dependency (task 2)	13	5.76
Structured fixed number of sequences (task 3)	1.8	0.47
Structured Output SVM (task 5)	13	5.76

Tabulka 2: Comparison of all classifiers.