STATISTICAL MACHINE LEARNING (WS2018)
SEMINAR 7

Assignment 1. Let $s_0, s_2, \ldots, s_{n-1}$ be K-valued random variables, where K is a finite set. Their joint probability distribution is a Markov model on a *cycle*

$$p(s) = \frac{1}{Z} \prod_{i=0}^{n-1} g_i(s_i, s_{i+1})$$

where indices $i+1$ are considered modulo n. The functions $g_i \colon K^2 \to \mathbb{R}_+$ are given and Z is a normalisation constant. Find an algorithm for searching the most probable realisation

$$s^* = \arg\max_{s \in K^n} p(s).$$

What complexity has it?

Assignment 2. Consider the class of $(\min, +)$-problems on graphs, which require to find the labelling

$$\boldsymbol{s}^* = \arg\min_{\boldsymbol{s} \in K^V} \sum_{i \in V} u_i(s_i) + \sum_{\{i,j\} \in E} u_{ij}(s_i, s_j), \tag{1}$$

where (V, E) is an undirected graph, K is a finite label set, $\boldsymbol{s} \colon V \to K$ is labelling of the nodes and $u_i \colon K \to \mathbb{R}$ and $u_{ij} \colon K^2 \to \mathbb{R}$ are given functions.
a) Prove that this class is NP-complete by reducing the maximum clique problem to it.
Hint: Suppose that the graph (V', E') is an input instance for the maximum clique problem. Consider the graph (V, E) with $V = V'$, $E = \overline{E'}$ and the label set $K = \{0, 1\}$. Find functions u_i and u_{ij} such that a labelling $\boldsymbol{s}$ is optimal if and only if it "encodes" a maximum clique.
b) Show that a $(\min, +)$-problem (1) can be solved approximately by α-expansions if the pairwise functions u_{ij} have the form

$$u_{ij}(k, k') = \beta_{ij} \mathbf{1}\{k \neq k'\} \ \text{ with } \ \beta_{ij} \geqslant 0\,.$$

Assignment 3. Consider a linear classifier $h \colon \mathcal{X} \times \mathcal{X} \to \mathcal{Y} \times \mathcal{Y}$ predicting a pair of labels $(y_1, y_2) \in \mathcal{Y} \times \mathcal{Y}$ from a pair of inputs $(x_1, x_2) \in \mathcal{X} \times \mathcal{X}$ based on the rule

$$h(x_1, x_2; \boldsymbol{\theta}) = \arg\max_{y_1 \in \mathcal{Y}, y_2 \in \mathcal{Y}} (\langle \boldsymbol{\phi}(x_1), \boldsymbol{w}_{y_1} \rangle + \langle \boldsymbol{\phi}(x_1), \boldsymbol{w}_{y_1} \rangle + g(y_1, y_2)) \tag{2}$$

where $\boldsymbol{\phi} \colon \mathcal{X} \to \mathbb{R}^n$ is a feature map, $\boldsymbol{w}_y \in \mathbb{R}^n$, $y \in \mathcal{Y}$, are vectors and $g \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a function. The vector $\boldsymbol{\theta} \in \mathbb{R}^{n|\mathcal{Y}| + |\mathcal{Y}|^2}$ encapsulates all parameters of the classifier, that is, the vectors $\{\boldsymbol{w}_y \in \mathbb{R}^n \mid y \in \mathcal{Y}\}$ and the function values $\{g(y, y') \in \mathbb{R} \mid y \in \mathcal{Y}, y' \in \mathcal{Y}\}$.

Let $\mathcal{T}^m = \{(x_1^j, x_2^j, y_1^j, y_2^j) \in (\mathcal{X} \times \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}) \mid j = 1, \ldots, m\}$ be a set of training examples. Describe a variant of the Perceptron algorithm that finds the parameters $\boldsymbol{\theta}$ such that the classifier (2) predicts all examples from $\mathcal{T}^m$ correctly, provided such parameters exist.