STATISTICAL MACHINE LEARNING (WS2019)
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?
Hint: Consider to use dynamic programing restricted to a single starting state.

Assignment 2. Consider a hidden Markov model

$$p(x, s) = p(s_1) \prod_{i=2}^{n} p(s_i \mid s_{i-1}) \prod_{i=1}^{n} p(x_i \mid s_i),$$

where $x = (x_1, \ldots, x_n)$ is a sequence of features and $s = (s_1, \ldots, s_n)$ is a sequence of hidden states, with values s_i from a finite set K. Given a sequence of of features x we want to predict the sequence of hidden states that has generated x.
a) Suppose we use the simple 0/1-loss $\ell(s, s') = \mathbb{1}\{s \not\equiv s'\}$. Prove that the optimal predictor $h(x)$ that minimises the expected loss

$$R(x, h) = \sum_{s \in K^n} p(x, s) \ell(s, h(x)),$$

is given by

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

b) Let us consider a more suitable loss – the Hamming distance between sequences s and s'

$$\ell(s, s') = \sum_{i=1}^{n} \mathbb{1}\{s_i \neq s'_i\}.$$

Show that the optimal predictor for this loss is given by

$$s_i^* = \arg\max_{k \in K} p(s_i = k, x),$$

i.e. predicting the sequence of most probable states.

Hint: Consider the expected loss for the Hamming distance, move the sum over the positions i outside of the summation over the sequences and analyse the resulting terms. Notice that the derivations in a) and b) are generic and do not presume that the model $p(x, s)$ for the sequences x, s is an HMM.
c*) The predictor in b) requires to compute the marginal probabilities $p(s_i = k, x)$ for all positions i and all states $k \in K$. Show that for an HMM they can be efficiently computed by performing dynamic matrix-vector multiplications from left to right and from right to left and combining the results.

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

$$h(x_1, x_2; \boldsymbol{\theta}) = \operatorname*{arg\,max}_{y_1 \in \mathcal{Y}, y_2 \in \mathcal{Y}} \big(\langle \boldsymbol{\phi}(x_1), \boldsymbol{w}_{y_1} \rangle + \langle \boldsymbol{\phi}(x_1), \boldsymbol{w}_{y_1} \rangle + g(y_1, y_2) \big) \tag{1}$$

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 (1) predicts all examples from $\mathcal{T}^m$ correctly, provided such parameters exist.
Hint: Try to express the condition under which the classifier (1) correctly predicts an example from the training data in terms of a system of linear inequalities.