8. Supervised learning of HMMs: Empirical risk minimisation

Given: i.i.d. training data $\mathcal{T} = \{(x^j, s^j) \mid x^j \in F^n, s^j \in K^n, j=1,\dots,m\}$
and the loss function $\ell(s,s') = \mathbb{1}\{s \neq s'\}$

Recall: optimal predictor $h: F^n \to K^n$ for 0/1 loss is

$$h_u(x) \in \operatorname*{argmax}_{s \in K^n} p_u(x,s)$$

Empirical risk minimisation:

$$\frac{1}{m}\sum_{j=1}^{m} \mathbb{1}\{s^j \neq h_u(x^j)\} \to \min_u$$

This task is not tractable because the objective function is piece-wise constant.

Special case: Suppose, $\exists\, u^*$ s.t. the empirical risk is zero.
How to find it? Conditions for u^*:

$$s^j \in \operatorname*{argmax}_{s \in K^n} p_{u^*}(x^j, s) \quad \forall j=1,\dots,m$$

or, equivalently

$$\langle \varphi(x^j,s^j), u^* \rangle > \langle \varphi(x^j,s), u^* \rangle \quad \forall s \neq s^j, \ \forall j=1,\dots,m$$

This is a system of linear inequalities $\Rightarrow$ perceptron algorithm
Start with arbitrary u and iterate

- find $\tilde{s}^j = \operatorname*{argmax}_{s \in K^n} \langle \varphi(x^j,s), u \rangle \quad j=1,\dots,m$

  This can be done by the algorithm in Sec. 4

- if for some j $\tilde{s}^j \neq s^j$, update u by

  $$u \to u + \varphi(x^j,s^j) - \varphi(x^j,\tilde{s}^j)$$

General case

Idea: overcome intractability by replacing the loss (as a function of u) by a convex upper bound. E.g. „margin rescaling" surrogate

$$\mathbb{1}\{s \neq h_u(x)\} \leq \max_{s' \in K^n} \left\{ \mathbb{1}\{s \neq s'\} + \langle \varphi(x,s') - \varphi(x,s), u \rangle \right\}$$

The approximation task reads

$$\frac{1}{m} \sum_{j=1}^{m} \max_{s \in K^n} \left\{ \mathbb{1}\{s \neq s^j\} + \langle \varphi(x^j,s) - \varphi(x^j,s^j), u \rangle \right\} \to \min_u$$

Solve by subgradient descent, cutting plane algorithm, ...

The inner optimisation tasks $\max_{s \in K^n} \{\dots\}$ are solved by the algorithm in Sec. 4.

<u>Remark 1</u> This approach is designated as „Structured Output SVM" and can be generalised for more complex losses as e.g. the Hamming distance.

9. Unsupervised learning: EM algorithm for HMMs

Given: i.i.d. training data $\mathcal{T} = \{x^j \in F^n \mid j = 1, \dots, m\}$

ML estimator: $$u^* \in \underset{u}{\operatorname{argmax}} \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \log \sum_{s \in K^n} p_u(x, s)$$

Recall EM algorithm

$$L(u) = \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \log \sum_{s \in K^n} \frac{\alpha(s|x)}{\alpha(s|x)} p_u(x, s),$$

where $\alpha(s|x) \geq 0$, $\sum_{s \in K^n} \alpha(s|x) = 1 \;\; \forall\, x \in \mathcal{T}$

Using concavity of log, we get a lower bound

$$L(u) \geq L_B(u, \alpha) = \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \sum_{s \in K^n} \alpha(s|x) \log \frac{p_u(x, s)}{\alpha(s|x)}$$

Equivalently

$$L_B(u, \alpha) = \mathbb{E}_{\mathcal{T}} \left[\log p_u(x) - D_{KL}(\alpha(s|x) \,\|\, p(s|x)) \right]$$

EM algorithm: Maximise $L_B(u, \alpha)$ by block-coordinate ascent w.r.t. α and u. Start with some $u^{(0)}$.

E-step set $\alpha^{(t)}(s|x) = p_{u^{(t)}}(s|x) \;\; \forall\, s \in K^n, \forall\, x \in \mathcal{T}$

M-step set

$$u^{(t+1)} \in \underset{u}{\operatorname{argmax}} \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \sum_{s \in K^n} \alpha^{(t)}(s|x) \log p_u(x, s)$$

Let us analyse the M-step for HMMs. The objective is

$$\frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \sum_{s \in K^n} \alpha^{(t)}(s|x) \langle \varphi(x, s), u \rangle - \log Z(u) \to \max_u$$

Denoting
$$\psi = \frac{1}{|T|} \sum_{x \in T} \sum_{s \in K^n} \alpha^{(t)}(s|x)\,\varphi(x,s)$$
we get
$$\langle \psi, u \rangle - \log Z(u) \to \max_u .$$

This is equivalent to the supervised learning task in Sec. 7. We know how to solve it, provided we can compute ψ.

<u>Computing ψ</u>:

For each $x \in T$ compute
$$\psi(x) = \sum_{s \in K^n} \alpha^{(t)}(s|x)\,\varphi(x,s) = \mathbb{E}_{p_{u^{(t)}}(s|x)}\,\varphi(x,s),$$
i.e. we have to compute posterior pairwise marginals $p(s_{i-1}, s_i \,|\, x)$ $\forall\, i = 2,\dots,n$ and $s_{i-1}, s_i \in K$. This can be done by an algorithm similar to the one discussed in Sec. 5

The components of ψ are then obtained by averaging the components of $\psi(x)$ over all $x \in T$, i.e. $\psi = \mathbb{E}_T\, \psi(x)$.

<u>Theorem 1</u> (w/o proof)

The sequence $L(u^{(t)})$ is monotonously increasing and the sequence $\alpha^{(t)}$ is convergent.

<u>Remark 1</u> The EM algorithm for HMMs is referred to as Baum-Welch algorithm.