Semi-probabilistic model

Probabilistic framework. Let the data be generated by a joint distribution of two random variables: a feature vector $\boldsymbol{x} \sim X$ and a class label $y \sim Y$ from a given parametric family ($\boldsymbol{\theta}$ is the parameter), with each observation being independent:

$$\boldsymbol{x}, y \sim f_{X,Y}(\boldsymbol{x}, y|\boldsymbol{\theta}) \sim (X \times Y)^{\ell}, \quad \boldsymbol{x} \in \mathbb{R}^k, y = 1..N \tag{1}$$

Given a training sample, we need to estimate the parameter values and build a predictive model:

$$\hat{y}(\boldsymbol{x}) = a_{\boldsymbol{\theta}}(\boldsymbol{x}) \tag{2}$$

Model. We will use the formalism of **semi-probabilistic models**, *i.e.*, we will consider X values as fixed and Y values as variables in the model. In other words, as a predictive model, we will build a probability distribution $f_Y(y|\boldsymbol{x}^*, \boldsymbol{\theta})$ for the random variable Y, parameterized by the data $\boldsymbol{x}^* \in X^{\ell}$ and the distribution shape parameters $\boldsymbol{\theta}$

Likelihood function. Let's write the joint distribution for all observed points as a product of independent distributions:

$$\begin{aligned} L(\boldsymbol{\theta}) &= \prod_{(\boldsymbol{x}^*, y^*) \in (X,Y)^{\ell}} \underbrace{f(\boldsymbol{x} = \boldsymbol{x}^*, y = y^*|\boldsymbol{\theta})}_{\text{likelihood}} \\ &= \prod_{(\boldsymbol{x}^*, y^*) \in (X,Y)^{\ell}} \frac{\overbrace{f_Y(y = y^*|\boldsymbol{x} = \boldsymbol{x}^*, \boldsymbol{\theta})}^{\text{predictive model } a_{\boldsymbol{\theta}}(\boldsymbol{x})}}{\underbrace{\cancel{f_X(\boldsymbol{x} = \boldsymbol{x}^*|\boldsymbol{\theta})}}_{\text{prior distribution}}} \end{aligned} \tag{3}$$

$$f(x, y|\theta) = \frac{f(x, y, \theta)}{f(\theta)} = \frac{f(y|x, \theta)/f(x, \theta)}{f(\theta)} = \frac{f(y|x, \theta)}{f(\theta) \cdot f(x, \theta)} = \frac{f(y|x, \theta)}{\cancel{f(x|\theta)}}$$

Applying the logarithm to the likelihood function, we obtain the log-likelihood function:

$$\ell(\boldsymbol{\theta}) = \sum_{(\boldsymbol{x}^*, y^*) \in (X,Y)^{\ell}} \ln \mathbb{P}(y = y^*|\boldsymbol{x} = \boldsymbol{x}^*, \boldsymbol{\theta}) \to \max_{\boldsymbol{\theta}} \tag{4}$$

Parameter estimation. Thus, we can find the parameter values $\boldsymbol{\theta}$ and use them in the conditional distribution for prediction:

$$a_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}') = \underbrace{f_Y\left(y|\boldsymbol{x} = \boldsymbol{x}', \boldsymbol{\theta} = \hat{\boldsymbol{\theta}}\right)}_{\text{distribution of } Y \text{ must be chosen}} \tag{5}$$

The prior distribution $f_X(\boldsymbol{x}|\boldsymbol{\theta})$ is canceled out from the product above.

- The maximum likelihood method considers the prior distribution of x unknown and

unimportant (unlike in MAP), focusing solely on the conditional distribution $\mathbb{P}[y|\boldsymbol{x}^*, \boldsymbol{\theta}]$ which serves as a model for building the algorithm $\hat{y} = a(x)$.

- Generally, when the chosen prior distribution is parameter-independent ($f_{\{X\}}(\boldsymbol{x}|\boldsymbol{\theta}) \equiv f_{\{X\}}(x)$),

it naturally cancels out since it remains constant for fixed $\boldsymbol{x}^*$ and does not depend on $\boldsymbol{\theta}$.

Cross-entropy loss. From summing over specific points $y^* \in Y^{\ell}$, we can transition to summing over all possible values $y' = 1..N$, assuming zero probability for point y^* belonging to another class $y' \neq y^*$ and defining $0 \cdot \ln 0 \equiv 0$ by definition.

$$\begin{aligned} \ell &= \sum_{\boldsymbol{x} \in X^{\ell}} \sum_{y' \in \operatorname{supp} Y} [\![y' = y^*]\!] \cdot \ln \mathbb{P}[y = y'|\boldsymbol{x} = \boldsymbol{x}^*, \boldsymbol{\theta}] \\ &\sim \sum_{\boldsymbol{x} \in X^{\ell}} \sum_{y' \in \operatorname{supp} Y} \underbrace{\mathbb{P}[y = y'|\boldsymbol{x} = \boldsymbol{x}^*, \boldsymbol{\theta}]}_{\text{smooth approximation of } [\![]\!]} \cdot \ln \mathbb{P}[y = y'|\boldsymbol{x} = \boldsymbol{x}^*, \boldsymbol{\theta}] \to \min_{\boldsymbol{\theta}} \end{aligned} \tag{6}$$

This is cross-entropy loss, which can be used if the model predicts **probabilities of belonging to each class**.