Feature Selection via Perfomance Prediction Model

Let we are given dataset $(\mathbf{X}, \mathbf{y})$, where $\mathbf{X} \in \mathbb{R}^{m \times n}$ is a design matrix, $\mathbf{y} \in \mathbb{R}^m$ is a target vector. The set $\mathcal{A} \subseteq \{1, \dots, n\}$ indicates subset of features. There is a correspondence between the set $\mathcal{A}$ and binary vectors $\mathbf{a} \in \mathbb{B}^n$:

$$\mathcal{A} = \{j : a_j = 1\}.$$

Function $f(\mathbf{x}, \mathbf{w}, \mathcal{A})$ predicts y given the object $\mathbf{x}$ and using only features from the set $\mathcal{A}$. We split our data into train $(\mathbf{X}_{\text{tr}}, \mathbf{y}_{\text{tr}})$ and test $(\mathbf{X}_{\text{te}}, \mathbf{y}_{\text{te}})$ parts. To measure the quality of the model f we introduce the error function $s(\mathbf{w}, \mathbf{X}, \mathbf{y}, \mathcal{A})$. On the training stage we find the optinal parameters $\mathbf{w}^*$

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} s(\mathbf{w}, \mathbf{X}_{\text{tr}}, \mathbf{y}_{\text{tr}}, \mathcal{A}).$$

Then we estimate the error on the test data $s(\mathbf{w}^*, \mathbf{X}_{\text{te}}, \mathbf{y}_{\text{te}}, \mathcal{A})$. The goal is to find the optimal feature subset $\mathcal{A}^*$, which minimizes the test error.

We propose the following procedure. Let create the dataset $\{(\mathbf{a}_i, s_i)\}_{i=1}^N$. We sample N vectors $\mathbf{a}$ from the binary cube $\mathbb{B}^n$. Then we evaluate the test error s_i for each feature subset $\mathbf{a}_i$. The idea is to change the discrete domain $\mathbb{B}^n$ to the continious one with smaller dimensionality $h < n$. We propose to embed feature indicator vector $\mathbf{a}$ into continious representation $\mathbf{u} \in \mathbb{R}^h$. This embedding is performed by encoder model $e(\mathbf{a}, \mathbf{w}_e)$. Then performance prediction model $p(\mathbf{u}, \mathbf{w}_p)$ tries to predict the test error s. The model has the form $s = p(\mathbf{u}, \mathbf{w}_p) = p(e(\mathbf{a}, \mathbf{w}_e), \mathbf{w}_p)$. We could use any loss function to estimate parameters $\mathbf{w}_e, \mathbf{w}_p$. For example it could be squared error

$$L_{error}(\mathbf{w}_e, \mathbf{w}_p, \mathbf{a}, s) = \|p(e(\mathbf{a}, \mathbf{w}_e), \mathbf{w}_p) - s\|^2 \to \min_{\mathbf{w}_e, \mathbf{w}_p}.$$

We also need the model to reconstruct the vector $\mathbf{a}$ from the continuous representation $\mathbf{u}$. This model is called decoder $\sigma(\mathbf{u}, \mathbf{w}_\sigma)$. We introduce the reconstruction loss as the cross-entropy between initial vector $\mathbf{a}$ and the output of the decoder σ

$$L_{rec}(\mathbf{w}_e, \mathbf{w}_\sigma, \mathbf{a}) = \sum_{i=1}^n a_i \log(\sigma_i) + (1 - a_i) \log(1 - \sigma_i) \to \min_{\mathbf{w}_e, \mathbf{w}_\sigma}.$$

Here $\sigma_i = \sigma(e(\mathbf{a}, \mathbf{w}_e), \mathbf{w}_\sigma)_i$.

$$\begin{tikzcd}
a \in \mathbb{B}^n \arrow[r, "e", "\text{encoder}"'] & u \in \mathbb{R}^h \arrow[r, "\sigma", "\text{decoder}"'] \arrow[d, "\text{perfomance prediction}"', "P"] & a \in \mathbb{B}^n \\
 & s \in \mathbb{R}^1 &
\end{tikzcd}$$

The final loss function is given by combination of the prediction performation loss and reconstruction loss

$$L = L_{error} + \alpha L_{rec}.$$

When the parameters $\mathbf{w}_e, \mathbf{w}_p, \mathbf{w}_\sigma$ are estimated we could find the most appropriate feature subset. We maximizes performance prediction model in the following way

$$\mathbf{u}^* = \arg\max_{\mathbf{u}} p(\mathbf{u}, \mathbf{w}_p).$$

To recover the feature vector $\mathbf{a}$ we use the decoder model

$$\mathbf{a}^* = \sigma(\mathbf{u}^*, \mathbf{w}_\sigma).$$