mlp-wd-1

The mlp-wd-1 methods implements a multilayer perceptron using weight decay. Fully connected feedforward networks with tanh units are trained using conjugate gradient. The number of hidden units is chosen using the following information argument: If each weight in the network takes on average r bits to specify and if training set, $\mathcal{T}$, contains $I_{\mathcal{T}}$ bits then the number of hidden units is chosen as:

$$n_h = \frac{I_{\mathcal{T}}/r - 1}{n_i + 2} \tag{1}$$

where a single output unit has been assumed and n_i is the number of input units. The information (in bits) of a training set is estimated as:

$$I_{\mathcal{T}} = 0.5 \log_2 \frac{\sigma_t^2}{\sigma_r^2}$$

where σ_t^2 is the variance of the targets and σ_r^2 the residual variance of a linear regression model.

We use $r = \{0.5, 1\}$. The training set is equally divided into $n = 4$ parts; one part is used as a validation set and the remainder for training. Training is repeated four times so that all parts are used once as the validation set. Conventional weight decay uses the cost function:

$$C(\alpha) = < E > + \alpha \sum_k w_k^2 \tag{2}$$

Where $< E >$ is the data misfit. We repeatedly use $\alpha = \{0,\ 0.008,\ 0.016,\ 0.032,\ 0.064,\ 0.125,\ 0.25,\ 0.5,\ 1,\ 2\}$ for each split of the data into training/validation set. Up to 50 epochs of conjugate gradient updates are used during the minimization. For each size of network and choice of α the total validation set prediction error $C(\alpha)$ over all splits is computed and an optimal weight decay factor chosen:

$$\alpha^* = \overset{min}{\alpha} \; C(\alpha)$$

The final network for prediction is trained using α^* and all the available training data.