Preliminaries: Classical coding theory

- Let $\mathbf{d}^i \in R^n$ be an input vector occurring with probability distribution P_d.
  Let $\mathbf{c}^i \in R^k$ be the corresponding k-dimensional transmitted vector with a probability distribution P_c.

- The receiver is concerned with the conditional probability $P(\mathbf{d}^i|\mathbf{c}^i)$ (low entropy of this quantity).

- Using Bayes rule

$$P(\mathbf{d}^i|\mathbf{c}^i) = \frac{P(\mathbf{c}^i|\mathbf{d}^i)}{P(\mathbf{c}^i)} P(\mathbf{d}^i).$$

Classical coding theory (continued)

- Optimal coding of $\mathbf{c}^i$ requires on average $-\sum_i P_c(\mathbf{c}^i) \log P_c(\mathbf{c}^i)$ bits.

- To this we add coding of the difference between the output and desired output (see below).

- Optimizing the code $\mathbf{c}$ subject to code length constraints is done by minimizing the relative entropy or the Kullback Leibler distance between the two probability distributions:

$$\begin{aligned} D(P_d \parallel P_c) &= \sum_i P_d(\mathbf{d}_i) \log \frac{P_d(\mathbf{d}_i)}{P_c(\mathbf{c}_i)} \\ &= E_{P_d}[\log(P_d) - \log(P_c)]. \end{aligned}$$

We ignored the coding scheme complexity!

Main deviation from classical information theory approach

- Under a Sender/Receiver setup:
  - We can not assume that the decoder in the receiver is arbitrarily complex.

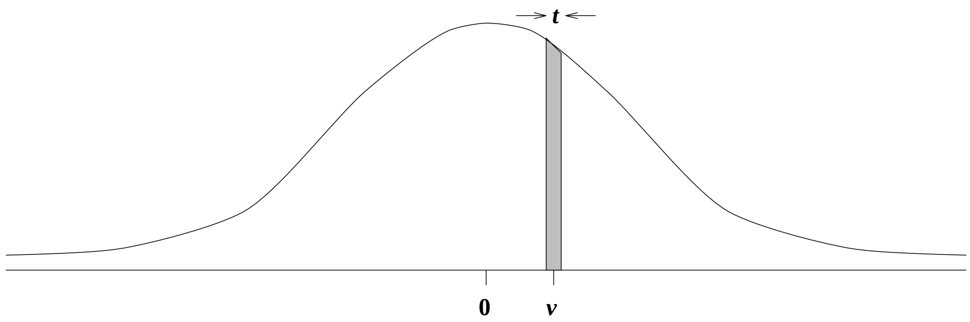
- We have to take learning into account by considering the probability of finding a good solution from every initial point.

Model Fit

- For output unit j, coding a real-valued number requires an infinite number of bits.

- We therefore have to assume that the output can be quantized into cells of width t.

- assuming further that the error is Normally distributed around the desired output **(unbiased model)**, we have



(From Hinton & van Camp, 1993)

Model Fit (Continued)

- The probability mass is approximated by t times the density at $(c_j - y_j)$.

- The code length of the residuals is therefore $-\log p(c_j - y_j)$.

- For a classification problem, the residual distribution can not be approximated by a Gaussian, however since it is low dimensional, density estimation is simple.

Model Fit (Continued)

- Under a Gaussian assumption e.g. for regression,

$$p(c_j - y_j) \simeq \frac{1}{\sqrt{2\pi}\sigma_j} t \, \mathrm{exp}^{-(c_j - y_j)^2 / 2\sigma_j^2}$$

- We code the data misfit using the posterior error distribution as it is already known to the sender (the network after training).

Coding model parameters

- During training we use the prior distribution on the weights to penalize our weight search.

- After training, we code the weights using the posterior distribution of the weights.

- The expected code length is

$$-\int_w p_w(w) \log p_w(w) dw.$$

- Under a mean zero Gaussian distribution assumption on the weights, the code length of the weights becomes $\frac{1}{2\sigma^2}\sum_i w_i^2$, namely, the standard weight decay penalty.

Combining it all together

- Given a weight vector w, the expected length of coding the residuals is
$$-\int_c p(c) \log p(c-y) dc.$$

- Note that the probability density of the output c is dependent on the particular parameter weight w.

- We shall approximate this probability by a single one!

- We now have to integrate the description length over the parameter distribution, namely
$$DL_{residuals} = -\int_w p(w) \int_c p(c) \log p(c-y) dc dw.$$

Remarks

- It is crucial to use the posterior distribution in the last term as it correctly represent the resulting state of the model after training.

- It is possible to constrain this distribution during training, for example to a mixture of Gaussians, as is done in Nowlan and Hinton (1992).

- The total description is given by:

$$\begin{aligned} DL &= -\int_w \int_c p(w)p(c)[\log p(c-y) \\ &\quad + \log p_w(w)]dcdw. \end{aligned}$$