Learning to Predict Probabilities

Consider target function/concept $c : X \rightarrow \{0, 1\}$ and training examples $D = \{\langle \mathbf{x}_d, t_d \rangle\}$ where $t_d = c(\mathbf{x}_d)$. For example,

- X denotes patients in terms of their symptoms and $c(\mathbf{x})$ is of value 1 if patient $\mathbf{x}$ survives disease, and 0 otherwise
- X denotes loan applicants in terms of their past credit history and $c(\mathbf{x})$ is of value 1 if loan applicant $\mathbf{x}$ repays loan, and 0 otherwise

Learn a neural network to output $P(c(\mathbf{x}) = 1)$ via the maximum likelihood hypothesis h_{ML} :

$$h_{\mathrm{ML}} = \arg\max_{h \in H} \sum_{d \in D} t_d \ln h(\mathbf{x}_d) + (1 - t_d) \ln(1 - h(\mathbf{x}_d))$$

Learning to Predict Probabilities

$$
\begin{aligned}
P(D|h) &= \prod_{d \in D} P(\mathbf{x}_d, t_d|h) = \prod_{d \in D} P(t_d|h, \mathbf{x}_d) P(\mathbf{x}_d) \\
P(t_d|h, \mathbf{x}_d) &= \begin{cases} h(\mathbf{x}_d) & \text{if } t_d = 1, \\ 1 - h(\mathbf{x}_d) & \text{if } t_d = 0; \end{cases} \\
&= h(\mathbf{x}_d)^{t_d} (1 - h(\mathbf{x}_d))^{1-t_d} \\
h_{\mathrm{ML}} &= \arg\max_{h \in H} p(D|h) \\
&= \arg\max_{h \in H} \prod_{d \in D} h(\mathbf{x}_d)^{t_d} (1 - h(\mathbf{x}_d))^{1-t_d} P(\mathbf{x}_d) \\
&= \arg\max_{h \in H} \prod_{d \in D} h(\mathbf{x}_d)^{t_d} (1 - h(\mathbf{x}_d))^{1-t_d} \\
&= \arg\max_{h \in H} \sum_{d \in D} t_d \ln h(\mathbf{x}_d) + (1 - t_d) \ln(1 - h(\mathbf{x}_d))
\end{aligned}
$$

Gradient Ascent to Maximize Likelihood in a Sigmoid Unit

$$
\begin{aligned}
U_D(h) &= \sum_{d \in D} t_d \ln h(\mathbf{x}_d) + (1 - t_d) \ln(1 - h(\mathbf{x}_d)) \\
\frac{\partial U_D}{\partial w_i} &= \sum_{d \in D} \frac{\partial U_D}{\partial h(\mathbf{x}_d)} \frac{\partial h(\mathbf{x}_d)}{\partial w_i} \\
&= \sum_{d \in D} \frac{\partial (t_d \ln h(\mathbf{x}_d) + (1 - t_d) \ln(1 - h(\mathbf{x}_d)))}{\partial h(\mathbf{x}_d)} \frac{\partial h(\mathbf{x}_d)}{\partial w_i} \\
&= \sum_{d \in D} \frac{t_d - h(\mathbf{x}_d)}{h(\mathbf{x}_d)(1 - h(\mathbf{x}_d))} h(\mathbf{x}_d)(1 - h(\mathbf{x}_d)) x_{id} \\
&= \sum_{d \in D} (t_d - h(\mathbf{x}_d)) x_{id}
\end{aligned}
$$

$$
w_i \quad \leftarrow \quad w_i + \Delta w_i \quad \text{where} \quad \Delta w_i = \eta \frac{\partial U_D}{\partial w_i}
$$

Minimum Description Length (MDL) Principle

Occam's razor. Prefer shortest hypothesis that fits the data

$$
\begin{aligned}
h_{\mathrm{MAP}} &= \arg\max_{h \in H} P(D|h)P(h) \\
&= \arg\max_{h \in H} \log_2 P(D|h) + \log_2 P(h) \\
&= \arg\min_{h \in H} \boxed{-\log_2 P(D|h)} \boxed{-\log_2 P(h)}
\end{aligned}
$$

Result of information theory. Optimal (shortest expected description length) code for a message with probability p is $-\log_2 p$ bits

- $-\log_2 P(h)$ is description length of h under optimal code
- $-\log_2 P(D|h)$ is description length of D given h under optimal code

Minimum Description Length (MDL) Principle

MDL. Select hypothesis that minimizes

$$h_{\mathrm{MDL}} = \arg\min_{h \in H} L_{C_1}(h) + L_{C_2}(D|h)$$

where $L_C(x)$ is description length of x under encoding C

Example. H = decision trees

- $L_{C1}(h)$ is #bits to describe tree h
- $L_{C2}(D|h)$ is #bits to describe D given h
  - $L_{C2}(D|h) = 0$ if examples classified perfectly by h. Otherwise, only misclassifications need to be described.
- By minimizing *length*(*tree*) & *length*(*misclassifications*(*tree*)), h_{MDL} trades off tree size for training errors > mitigate overfitting

Most Probable Classification of New Instances

Given new instance **x**, what is its most probable classification given the training data D?

h_{MAP} is the most probable hypothesis, but not the most probable classification!

Example. Consider H with 3 possible hypotheses:

$$P(h_1|D) = .4 \quad P(h_2|D) = .3 \quad P(h_3|D) = .3 \ .$$

Suppose that new instance **x** is given and

$$h_1(\mathbf{x}) = + \quad h_2(\mathbf{x}) = - \quad h_3(\mathbf{x}) = - \ .$$

What is the most probable classification of **x**?

Bayes-Optimal Classifier

Bayes-optimal classification.

$$\arg\max_{t \in T} P(t|D) = \arg\max_{t \in T} \sum_{h \in H} P(t|h)P(h|D)$$

Example (cont'd). Let $T = \{+, -\}$. Then,

$$P(h_1|D) = .4 \quad P(-|h_1) = 0 \quad P(+|h_1) = 1$$

$$P(h_2|D) = .3 \quad P(-|h_2) = 1 \quad P(+|h_2) = 0$$

$$P(h_3|D) = .3 \quad P(-|h_3) = 1 \quad P(+|h_3) = 0$$

$$\sum_{h \in H} P(+|h)P(h|D) = \qquad \sum_{h \in H} P(-|h)P(h|D) =$$

$$\arg\max_{t \in \{+,-\}} \sum_{h \in H} P(t|h)P(h|D) =$$

Gibbs Classifier

Bayes-optimal classifier provides best performance but is computationally costly if H is large.

Gibbs algorithm.

- Sample a hypothesis h from posterior belief $P(h|D)$
- Use h to classify new instance $\mathbf{x}$

Surprising result. Supposing target concepts are sampled from some prior over H, expected misclassification error of Gibbs classifier is at most twice that of Bayes-optimal classifier.

Concept learning. Supposing target concepts are sampled from uniform prior over H, a hypothesis is sampled from uniform prior over VS and its expected misclassification error is no worse than twice that of Bayes-optimal classifier.