6. Probabilistic models

References
C. Bishop. Pattern Recognition and Machine Learning. Sect. 4.2, 4.3

Approaches that allows to estimate probability that given an instance we have a certain classes.

Two families of models:
Generative: estimate P(x|Ci) and then compute P(Ci |x) with Bayes
Discriminative: estimate P(Ci |x) directly

6.1 Probabilistic generative models

$$P(C_1|\mathbf{x}) = \frac{P(\mathbf{x}|C_1)P(C_1)}{P(\mathbf{x})} = \frac{P(\mathbf{x}|C_1)P(C_1)}{P(\mathbf{x}|C_1)P(C_1) + P(\mathbf{x}|C_2)P(C_2)}$$
$$= \frac{1}{1+\exp(-a)} = \sigma(a)$$

with:

$$a = \ln \frac{p(\mathbf{x}|C_1)P(C_1)}{p(\mathbf{x}|C_2)P(C_2)}$$

and

$$\sigma(a) = \frac{1}{1+\exp(-a)} \text{ the } \textit{sigmoid function.}$$

Multi-class

$$P(C_k|\mathbf{x}) = \frac{P(\mathbf{x}|C_k)P(C_k)}{\sum_j P(\mathbf{x}|C_j)P(C_j)} = \frac{exp(a_k)}{\sum_j exp(a_j)}$$

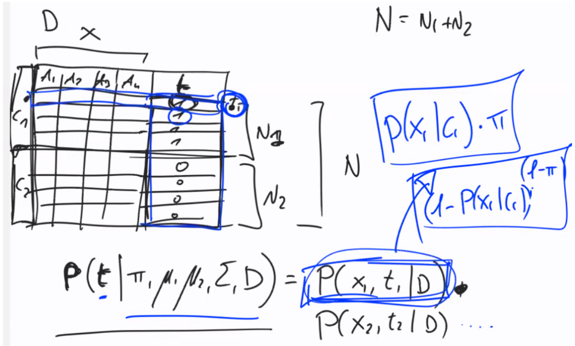
(normalized exponential or softmax function)

with $a_k = \ln P(\mathbf{x}|C_k)P(C_k)$

6.1.1 Maximum likelihood

Likelihood function

$$P(\mathbf{t}|\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}, D) = \prod^{N} [\overbrace{\pi}^{P(C_1)\,\cdot}\overbrace{\mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_1, \boldsymbol{\Sigma})}^{P(x|C_1)}]^{t_n} [\overbrace{(1-\pi)}^{P(C_2)}\overbrace{\mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_2, \boldsymbol{\Sigma})}^{P(x|C_2)}]^{(1-t_n)}$$



For 2 classes, we obtain

$$\pi = \frac{N_1}{N}$$

→ # of class 1.
→ Total

$$\boldsymbol{\mu}_1 = \frac{1}{N_1}\sum_{n=1}^{N} t_n \mathbf{x}_n \qquad \boldsymbol{\mu}_2 = \frac{1}{N_2}\sum_{n=1}^{N}(1-t_n)\mathbf{x}_n$$

$$\boldsymbol{\Sigma} = \frac{N_1}{N}S_1 + \frac{N_2}{N}S_2$$

with $S_i = \frac{1}{N_i}\sum_{n\in C_i}(\mathbf{x}_n - \boldsymbol{\mu}_i)(\mathbf{x}_n - \boldsymbol{\mu}_i)^T$, $i = 1, 2$

Posterior distributions with parametric models:

For two classes

$$P(C_1|\mathbf{x}) = \sigma(a)$$

For $k \geq 2$ classes

$$P(C_i|\mathbf{x}) = \frac{exp(a_k)}{\sum_j exp(a_j)}$$

$$a_k = \mathbf{w}^T\mathbf{x} + w_0$$

$$\mathbf{w}^T\mathbf{x} + w_0 = (w_0\ \mathbf{w})\begin{pmatrix} 1 \\ \mathbf{x} \end{pmatrix}$$

$$\tilde{\mathbf{w}} = \begin{pmatrix} w_0 \\ \mathbf{w} \end{pmatrix}, \tilde{\mathbf{x}} = \begin{pmatrix} 1 \\ \mathbf{x} \end{pmatrix}$$

$$a_k = \mathbf{w}^T\mathbf{x} + w_0 = \tilde{\mathbf{w}}^T\tilde{\mathbf{x}}$$

Likelihood for a parametric model $\mathcal{M}_\Theta$: $P(\mathbf{t}|\Theta, D)$, $D = \langle \mathbf{X}, \mathbf{t} \rangle$

Maximum likelihood solution:

$$\Theta^* = \underset{\Theta}{\mathrm{argmax}} \ln P(\mathbf{t}|\Theta, \mathbf{X})$$

When $\mathcal{M}_\Theta$ belongs to the exponential family, likelihood $P(\mathbf{t}|\Theta, \mathbf{X})$ can be expressed in the form $P(\mathbf{t}|\tilde{\mathbf{w}}, \mathbf{X})$, with maximum likelihood

$$\tilde{\mathbf{w}}^* = \underset{\tilde{\mathbf{w}}}{\mathrm{argmax}} \ln P(\mathbf{t}|\tilde{\mathbf{w}}, \mathbf{X})$$

6.2 Probabilistic discriminative models

Estimate dirrectly

$$P(C_i|\tilde{\mathbf{x}}, D) = \frac{exp(a_k)}{\sum_j exp(a_j)} \qquad a_k = \tilde{\mathbf{w}}^T\tilde{\mathbf{x}}$$

with maximum likelihood

Max the likelihood.

$$\tilde{\mathbf{w}}^* = \underset{\tilde{\mathbf{w}}}{\mathrm{argmax}} \ln P(\mathbf{t}|\tilde{\mathbf{w}}, \mathbf{X})$$

without estimating the model parameters.

Simplified notation (dataset omitted): $P(\mathbf{t}|\tilde{\mathbf{w}})$

6.2.1 Logistic regression

For 2 classes, likelihood function:

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}(1 - y_n)^{1-t_n}$$

with $y_n = p(C_1|\mathbf{x}_n) = \sigma(\mathbf{w}^T\mathbf{x}_n)$

Note:tn: value in the data set corresponding toxn,yn: posterior prediction of the current model wforxn.

Cross-entropy error function:

$$E(\mathbf{w}) \equiv -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^{N}[t_n \ln y_n + (1 - t_n)\ln(1 - y_n)]$$

Solution concept: solve the optimization problem

$$\mathbf{w}^* = \underset{\mathbf{w}}{\mathrm{argmin}}\, E(\mathbf{w})$$

To minimize error we apply Newton-Raphson:

$$\nabla E(\mathbf{w}) = \sum_{n=1}^{N} (y_n - t_n)\mathbf{x}_n$$

Gradient descent step

$$\mathbf{w} \leftarrow \mathbf{w} - \boldsymbol{H}^{-1}\nabla E(\mathbf{w})$$

$\boldsymbol{H} = \nabla\nabla E(\mathbf{w})$ is the Hessian matrix of $E(\mathbf{w})$ (second derivatives with respect to $\mathbf{w}$).

Given

$\tilde{\mathbf{X}} = \begin{pmatrix} \tilde{\mathbf{x}}_1^T \\ \dots \\ \tilde{\mathbf{x}}_N^T \end{pmatrix} \quad \mathbf{t} = \begin{pmatrix} t_1 \\ \dots \\ t_N \end{pmatrix},$

$\mathbf{y}(\tilde{\mathbf{w}}) = (y_1, \dots, y_n)^T$ posterior predictions of model $\tilde{\mathbf{w}}$

$\boldsymbol{R}(\tilde{\mathbf{w}})$: diagonal matrix with $R_{nn} = y_n(1 - y_n)$

we have

$$\nabla E(\tilde{\mathbf{w}}) = \tilde{\mathbf{X}}^T(\mathbf{y}(\tilde{\mathbf{w}}) - \mathbf{t})$$

$$\boldsymbol{H}(\tilde{\mathbf{w}}) = \nabla\nabla E(\tilde{\mathbf{w}}) = \sum^{N} y_n(1 - y_n)\tilde{\mathbf{x}}_n\tilde{\mathbf{x}}_n^T = \tilde{\mathbf{X}}^T\boldsymbol{R}(\tilde{\mathbf{w}})\tilde{\mathbf{X}}$$

Iterative method:

1. Initialize $\mathbf{w}$
2. Repeat until termination condition

$$\mathbf{w} \leftarrow \mathbf{w} - (\mathbf{X}^T\boldsymbol{R}\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{y} - \mathbf{t})$$

Multiclass:

K **classes**

$$P(C_k|\tilde{\mathbf{x}}) = \frac{exp(a_k)}{\sum_j exp(a_j)} \qquad a_k = \tilde{\mathbf{w}}_k^T \tilde{\mathbf{x}} \qquad k = 1, \dots, K$$

$$\tilde{\mathbf{X}} = \begin{pmatrix} \tilde{\mathbf{x}}_1^T \\ \dots \\ \tilde{\mathbf{x}}_N^T \end{pmatrix} \qquad \mathbf{T} = \begin{pmatrix} t_1^T \\ \dots \\ t_N^T \end{pmatrix} \text{ 1-of-}K \text{ encoding of labels}$$

$\mathbf{y}_n^T = (y_{n1} \dots y_{nK})^T$ posterior prediction of $\tilde{\mathbf{x}}_n$ for model $\tilde{\mathbf{w}}_1, \dots, \tilde{\mathbf{w}}_K$

$$\mathbf{Y}(\tilde{\mathbf{w}}_1, \dots, \tilde{\mathbf{w}}_K) = \begin{pmatrix} \mathbf{y}_1^T \\ \dots \\ \mathbf{y}_N^T \end{pmatrix} \text{ posterior predictions of model } \tilde{\mathbf{w}}_1, \dots, \tilde{\mathbf{w}}_K$$

Discriminative model

$$P(\mathbf{T}|\tilde{\mathbf{w}}_1, \dots, \tilde{\mathbf{w}}_K) = \prod_{n=1}^{N} \prod_{k=1}^{K} P(C_k|\tilde{\mathbf{x}}_n)^{t_{nk}} = \prod_{n=1}^{N} \prod_{k=1}^{K} y_{nk}^{t_{nk}}$$

with $y_{nk} = \mathbf{Y}[n,k]$ and $t_{nk} = \mathbf{T}[n,k]$.

Cross-entropy error function

$$E(\tilde{\mathbf{w}}_1, \dots \tilde{\mathbf{w}}_K) = -\ln P(\mathbf{T}|\tilde{\mathbf{w}}_1, \dots \tilde{\mathbf{w}}_K) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln y_{nk}$$

Iterative algorithm

gradient $\nabla_{\tilde{\mathbf{w}}_j} E(\tilde{\mathbf{w}}_1, \dots \tilde{\mathbf{w}}_K) = \dots$

Hessian matrix $\nabla_{\tilde{\mathbf{w}}_k} \nabla_{\tilde{\mathbf{w}}_j} E(\tilde{\mathbf{w}}_1, \dots \tilde{\mathbf{w}}_K) = \dots$

SUMMARY

Given a target function $f : X \to C$, and data set D

assume a parametric model for the posterior probability $P(C_k|\tilde{\mathbf{x}}, \tilde{\mathbf{w}})$
$\sigma(\tilde{\mathbf{w}}^T \tilde{\mathbf{x}})$ (2 classes) or $\frac{exp(\tilde{\mathbf{w}}_k^T \tilde{\mathbf{x}})}{\sum_{j=1}^{K} exp(\tilde{\mathbf{w}}_j^T \tilde{\mathbf{x}})}$ (k classes)

Define an error function $E(\tilde{\mathbf{w}})$ (negative log likelihood)

Solve the optimization problem

$$\tilde{\mathbf{w}}^* = \underset{\tilde{\mathbf{w}}}{\operatorname{argmin}}\, E(\tilde{\mathbf{w}})$$

Classify new sample $\tilde{\mathbf{x}}'$ as C_{k^*} where $k^* = \operatorname{argmax}_{k=1,\dots,K} P(C_k|\tilde{\mathbf{x}}', \tilde{\mathbf{w}}^*)$