Probabilistic classification - generative models

Course of Machine Learning
Master Degree in Computer Science
University of Rome "Tor Vergata"
a.a. 2020-2021

Giorgio Gambosi

Naive Bayes classifiers recap

A *language model* is a (categorical) probability distribution on a vocabulary of terms (possibly, all words which occur in a large collection of documents).

Use

A language model can be applied to predict (generate) the next term occurring in a text. The probability of occurrence of a term is related to its information content and is at the basis of a number of information retrieval techniques.

Hypothesis

It is assumed that the probability of occurrence of a term is independent from the preceding terms in a text (*bag of words* model).

Bayesian classifiers

A language model can be applied to derive document classifiers into two or more classes through Bayes' rule.

- given two classes C_1, C_2, assume that, for any document d, the probabilities $p(C_1|d)$ and $p(C_2|d)$ are known: then, d can be assigned to the class with higher probability
- how to derive $p(C_k|d)$ for any document, given a collection $\mathcal{C}_1$ of documents known to belong to C_1 and a similar collection $\mathcal{C}_2$ for C_2? Apply Bayes' rule:

  $$p(C_k|d) \propto p(d|C_k)p(C_k)$$

  the evidence $p(d)$ is the same for both classes, and can be ignored.
- we have still the problem of computing $p(C_k)$ and $p(d|C_k)$ from $\mathcal{C}_1$ and $\mathcal{C}_2$

Bayesian classifiers

Computing $p(C_k)$

The prior probabilities $p(C_k)$ $(k = 1, 2)$ can be easily estimated from $\mathcal{C}_1, \mathcal{C}_2$: for example, by applying ML, we obtain

$$p(C_k) = \frac{|\mathcal{C}_1|}{|\mathcal{C}_1| + |\mathcal{C}_2|}$$

Naive bayes classifiers

Computing $p(d|C_k)$

For what concerns the likelihoods $p(d|C_k)$ $(k = 1, 2)$, we observe that d can be seen, according to the bag of words assumption, as a multiset of n_d terms

$$d = \{\bar{t}_1, \bar{t}_2, \ldots, \bar{t}_{n_d}\}$$

By applying the product rule, it results

$$\begin{aligned}p(d|C_k) &= p(\bar{t}_1, \ldots, \bar{t}_{n_d}|C_k)\\ &= p(\bar{t}_1|C_k)p(\bar{t}_2|\bar{t}_1, C_k)\cdots p(\bar{t}_{n_d}|\bar{t}_1, \ldots, \bar{t}_{n_d-1}, C_k)\end{aligned}$$

Naive bayes classifiers

The naive Bayes assumption

Computing $p(d|C_k)$ is much easier if we assume that terms are pairwise conditionally independent, given the class C_k, that is, for $i, j = 1 \ldots, n_d$ and $k = 1, 2$,

$$p(\bar{t}_i, \bar{t}_j|C_k) = p(\bar{t}_i|C_k)p(\bar{t}_2|C_k)$$

as, a consequence,

$$p(d|C_k) = \prod_{j=1}^{n_d} p(\bar{t}_j|C_k)$$

that is, we model the document as a set of samples from a categorical distribution (the language model): ML is applied to select the best categorical distribution (class)

Language models and NB classifiers

The categorical distributions $p(\bar{t}_j|C_k)$ have been derived for C_1 and C_2, respectively from documents in $\mathcal{C}_1$ and $\mathcal{C}_2$.

Generative models

- Classes are modeled by suitable conditional distributions $p(\mathbf{x}|C_k)$ (language models in the previous case): it is possible to sample from such distributions to generate random documents statistically equivalent to the documents in the collection used to derive the model.
- Bayes' rule allows to derive $p(C_k|\mathbf{x})$ given such models (and the prior distributions $p(C_k)$ of classes)
- We may derive the parameters of $p(\mathbf{x}|C_k)$ and $p(C_k)$ from the dataset, for example through maximum likelihood estimation
- Classification is performed by comparing $p(C_k|\mathbf{x})$ for all classes

Deriving posterior probabilities

- Let us consider the binary classification case and observe that

$$p(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1) + p(\mathbf{x}|C_2)p(C_2)} = \frac{1}{1 + \frac{p(\mathbf{x}|C_2)p(C_2)}{p(\mathbf{x}|C_1)p(C_1)}}$$

- Let us define

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

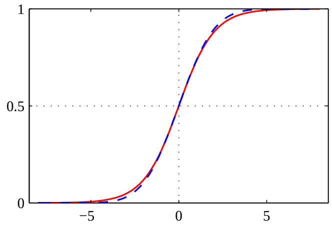
  that is, a is the log of the ratio between the posterior probabilities (*log odds*)

- We obtain that

$$p(C_1|\mathbf{x}) = \frac{1}{1 + e^{-a}} = \sigma(a) \qquad p(C_2|\mathbf{x}) = 1 - \frac{1}{1 + e^{-a}} = \frac{1}{1 + e^{a}}$$

- $\sigma(x)$ is the *logistic function* or (*sigmoid*)

Sigmoid



Useful properties of the sigmoid

- $\sigma(-x) = 1 - \sigma(x)$
- $\dfrac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x))$

Deriving posterior probabilities

- In the case $K > 2$, the general formula holds

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_j p(\mathbf{x}|C_j)p(C_j)}$$

- Let us define, for each $k = 1, \ldots, K$

$$a_k(\mathbf{x}) = \log(p(\mathbf{x}|C_k)p(C_k)) = \log p(C_k|\mathbf{x}) + \log p(C_k)$$

- Then, we may write

$$p(C_k|\mathbf{x}) = \frac{e^{a_k}}{\sum_j e^{a_j}} = s(a_k)$$

- $s(\mathbf{x})$ is the *softmax* function (or *normalized exponential*) and it can be seen as an extension of the sigmoid to the case $K > 2$
- $s(\mathbf{x})$ can be seen as a smoothed version of the maximum:
  if $a_k \gg a_j$ for all $j \neq k$, then $s(a_k) \simeq 1$ and $s(a_j) \simeq 0$ for all $j \neq k$

Gaussian discriminant analysis

In Gaussian discriminant analysis (GDA) all class conditional distributions $p(\mathbf{x}|C_k)$ are assumed gaussians. This implies that the corresponding posterior distributions $p(C_k|\mathbf{x})$ can be easily derived.

Hypothesis

All distributions $p(\mathbf{x}|C_k)$ have same covariance matrix Σ, of size $D \times D$. Then,

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \text{exp}\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\right)$$

Binary case

If $K = 2$,

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

where

$$
\begin{aligned}
a(\mathbf{x}) &= \log \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)} \\
&= \log \frac{\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)\right)p(C_1)}{\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)\right)p(C_2)} \\
&= \frac{1}{2}(\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 - \mathbf{x}^T\Sigma^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_2^T\Sigma^{-1}\mathbf{x}) - \\
&\quad - \frac{1}{2}(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \mathbf{x}^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_1^T\Sigma^{-1}\mathbf{x}) + \log\frac{p(C_1)}{p(C_2)}
\end{aligned}
$$

Binary case

Observe that the results of all products involving Σ^{-1} are scalar, hence, in particular

$$
\begin{aligned}
\mathbf{x}^T\Sigma^{-1}\boldsymbol{\mu}_1 &= \boldsymbol{\mu}_1^T\Sigma^{-1}\mathbf{x} \\
\mathbf{x}^T\Sigma^{-1}\boldsymbol{\mu}_2 &= \boldsymbol{\mu}_2^T\Sigma^{-1}\mathbf{x}
\end{aligned}
$$

Then,

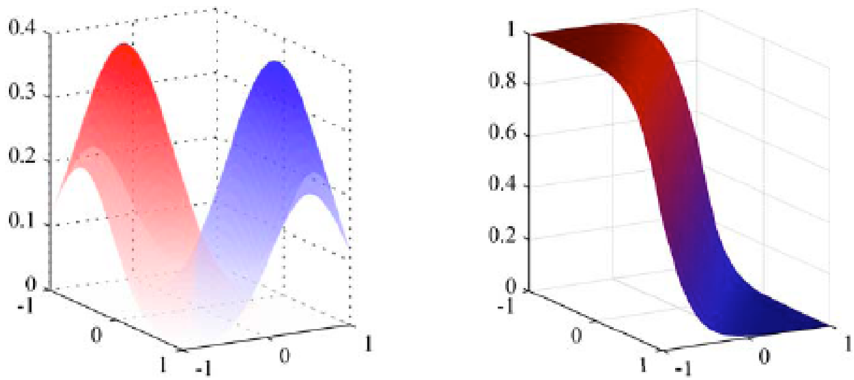
$$
a(\mathbf{x}) = \frac{1}{2}(\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1) + (\boldsymbol{\mu}_1^T\Sigma^{-1} - \boldsymbol{\mu}_2^T\Sigma^{-1})\mathbf{x} + \log\frac{p(C_1)}{p(C_2)} = \mathbf{w}^T\mathbf{x} + w_0
$$

with

$$
\begin{aligned}
\mathbf{w} &= \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \\
w_0 &= \frac{1}{2}(\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1) + \log\frac{p(C_1)}{p(C_2)}
\end{aligned}
$$

$p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$ is computed by applying a non-linear function to a linear combination of the features (*generalized linear model*)

Example



Left, the class conditional distributions $p(\mathbf{x}|C_1), p(\mathbf{x}|C_2)$, gaussians with $D = 2$. Right the posterior distribution of C_1, $p(C_1|\mathbf{x})$ with sigmoidal slope.

Discriminant function

The discriminant function can be obtained by the condition $p(C_1|\mathbf{x}) = p(C_2|\mathbf{x})$, that is, $\sigma(a(\mathbf{x})) = \sigma(-a(\mathbf{x}))$.

This is equivalent to $a(\mathbf{x}) = -a(\mathbf{x})$ and to $a(\mathbf{x}) = 0$. As a consequence, it results

$$
\mathbf{w}^T\mathbf{x} + w_0 = 0
$$

or

$$
\Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\mathbf{x} + \frac{1}{2}(\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1) + \log\frac{p(C_2)}{p(C_1)} = 0
$$

Simple case: $\Sigma = \lambda \mathbf{I}$ (that is, $\sigma_{ii} = \lambda$ for $i = 1, \ldots, d$). In this case, the discriminant function is

$$2(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)\mathbf{x} + ||\boldsymbol{\mu}_1||^2 - ||\boldsymbol{\mu}_2||^2 + 2\lambda \log \frac{p(C_2)}{p(C_1)} = 0$$

Multiple classes

In this case, we refer to the softmax function:

$$p(C_k|\mathbf{x}) = s(a_k(\mathbf{x}))$$

where $a_k(\mathbf{x}) = \log(p(\mathbf{x}|C_k)p(C_k))$.

By the above considerations, it easily turns out that

$$a_k(\mathbf{x}) = \frac{1}{2}\left(\boldsymbol{\mu}_k^T \Sigma^{-1}\mathbf{x} - \boldsymbol{\mu}_k^T \Sigma^{-1}\boldsymbol{\mu}_k\right) + \log p(C_k) - \frac{d}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| = \mathbf{w}_k^T\mathbf{x} + w_{0k}$$

Again, $p(C_k|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$ is computed by applying a non-linear function to a linear combination of the features (*generalized linear model*)

Multiple classes

Decision boundaries corresponding to the case when there are two classes C_j, C_k such that the corresponding posterior probabilities are equal, and larger than the probability of any other class. That is,

$$p(C_k|\mathbf{x}) = p(C_j|\mathbf{x}) \qquad\qquad p(C_i|\mathbf{x}) < p(C_k|\mathbf{x}) \qquad i \neq j, k$$

hence

$$e^{a_k(\mathbf{x})} = e^{a_j(\mathbf{x})} \qquad\qquad e^{a_i(\mathbf{x})} < e^{a^k(\mathbf{x})} \qquad i \neq j, k$$

that is,

$$a_k(\mathbf{x}) = a_j(\mathbf{x}) \qquad\qquad a_i(\mathbf{x}) < a^k(\mathbf{x}) \qquad i \neq j, k$$

As shown, this implies that boundaries are linear.

General covariance matrices, binary case

The class conditional distributions $p(\mathbf{x}|C_k)$ are gaussians with different covariance matrices

$$\begin{aligned}
a(\mathbf{x}) &= \log \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)} \\
&= \log \frac{\mathit{exp}\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1)\right)}{\mathit{exp}\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_2)^T\Sigma_2^{-1}(\mathbf{x} - \boldsymbol{\mu}_2)\right)} + \frac{1}{2}\log\frac{|\Sigma_2|}{|\Sigma_1|} + \log\frac{p(C_1)}{p(C_2)} \\
&= \frac{1}{2}\left((\mathbf{x} - \boldsymbol{\mu}_2)^T\Sigma_2^{-1}(\mathbf{x} - \boldsymbol{\mu}_2) - (\mathbf{x} - \boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1)\right) + \frac{1}{2}\log\frac{|\Sigma_2|}{|\Sigma_1|} + \log\frac{p(C_1)}{p(C_2)}
\end{aligned}$$

General covariance matrices, binary case

By applying the same considerations, the decision boundary turns out to be

$$\left((\mathbf{x} - \boldsymbol{\mu}_2)^T\Sigma_2^{-1}(\mathbf{x} - \boldsymbol{\mu}_2) - (\mathbf{x} - \boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{x} - \boldsymbol{\mu}_1)\right) + \log\frac{|\Sigma_2|}{|\Sigma_1|} + 2\log\frac{p(C_1)}{p(C_2)} = 0$$

Classes are separated by a (at most) quadratic surface.

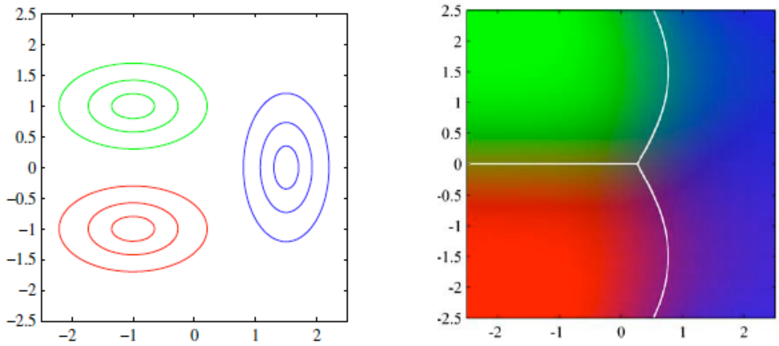
General covariance, multiple classe

It can be proved that boundary surfaces are at most quadratic.

Example

Left: 3 classes, modeled by gaussians with different covariance matrices.

Right: posterior distribution of classes, with boundary surfaces.



GDA and maximum likelihood

The class conditional distributions $p(\mathbf{x}|C_k)$ can be derived from the training set by maximum likelihood estimation.

For the sake of simplicity, assume $K = 2$ and both classes share the same Σ.

It is then necessary to estimate $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$, Σ, and $\pi = p(C_1)$ (clearly, $p(C_2) = 1 - \pi$).

GDA and maximum likelihood

Training set $\mathcal{T}$: includes n elements $(\mathbf{x}_i, t_i)$, with

$$t_i = \begin{cases} 0 & \text{se } \mathbf{x}_i \in C_2 \\ 1 & \text{se } \mathbf{x}_i \in C_1 \end{cases}$$

If $\mathbf{x} \in C_1$, then $p(\mathbf{x}, C_1) = p(\mathbf{x}|C_1)p(C_1) = \pi \cdot \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \Sigma)$

If $\mathbf{x} \in C_2$, $p(\mathbf{x}, C_2) = p(\mathbf{x}|C_2)p(C_2) = (1 - \pi) \cdot \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_2, \Sigma)$

The likelihood of the training set $\mathcal{T}$ is

$$L(\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma|\mathcal{T}) = \prod_{i=1}^{n} (\pi \cdot \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma))^{t_i}((1 - \pi) \cdot \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_2, \Sigma))^{1-t_i}$$

GDA and maximum likelihood

The corresponding log likelihood is

$$\begin{aligned} l(\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma|\mathcal{T}) = & \sum_{i=1}^{n} (t_i \log \pi + t_i \log(\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma))) + \\ & + \sum_{i=1}^{n} ((1 - t_i) \log(1 - \pi) + (1 - t_i) \log(\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_2, \Sigma))) \end{aligned}$$

Its derivative wrt π is

$$\frac{\partial l}{\partial \pi} = \frac{\partial}{\partial \pi} \sum_{i=1}^{n} (t_i \log \pi + (1 - t_i) \log(1 - \pi)) = \sum_{i=1}^{n} \left(\frac{t_i}{\pi} - \frac{(1 - t_i)}{1 - \pi} \right) = \frac{n_1}{\pi} - \frac{n_2}{1 - \pi}$$

which is equal to 0 for

$$\pi = \frac{n_1}{n}$$

GDA and maximum likelihood

The maximum wrt $\boldsymbol{\mu}_1$ (and $\boldsymbol{\mu}_2$) is obtained by computing the gradient

$$\frac{\partial l}{\partial \boldsymbol{\mu}_1} = \frac{\partial}{\partial \boldsymbol{\mu}_1} \sum_{i=1}^{n} t_i \log(\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma)) = \cdots = \Sigma^{-1} \sum_{i=1}^{n} t_i(\mathbf{x}_i - \boldsymbol{\mu}_1)$$

As a consequence, we have $\frac{\partial l}{\partial \boldsymbol{\mu}_1} = 0$ for

$$\sum_{i=1}^n t_i \mathbf{x}_i = \sum_{i=1}^n t_i \boldsymbol{\mu}_1$$

hence, for

$$\boldsymbol{\mu}_1 = \frac{1}{n_1} \sum_{\mathbf{x}_i \in C_1} \mathbf{x}_i$$

GDA and maximum likelihood

Similarly, $\frac{\partial l}{\partial \boldsymbol{\mu}_2} = 0$ for

$$\boldsymbol{\mu}_2 = \frac{1}{n_2} \sum_{\mathbf{x}_i \in C_2} \mathbf{x}_i$$

GDA and maximum likelihood

Maximizing the log-likelihood wrt Σ provides

$$\Sigma = \frac{n_1}{n}\mathbf{S}_1 + \frac{n_2}{n}\mathbf{S}_2$$

where

$$\mathbf{S}_1 = \frac{1}{n_1} \sum_{\mathbf{x}_i \in C_1} (\mathbf{x}_i - \boldsymbol{\mu}_1)(\mathbf{x}_i - \boldsymbol{\mu}_1)^T$$

$$\mathbf{S}_2 = \frac{1}{n_2} \sum_{\mathbf{x}_i \in C_2} (\mathbf{x}_i - \boldsymbol{\mu}_2)(\mathbf{x}_i - \boldsymbol{\mu}_2)^T$$

and let

$$\mathbf{S} = \frac{n_1}{n}\mathbf{S}_1 + \frac{n_2}{n}\mathbf{S}_2$$

GDA: discrete features

- In the case of d discrete (for example, binary) features we may apply the Naive Bayes hypothesis (independence of features, given the class)
- Then, we may assume that, for any class C_k, the value of the i-th feature is sampled from a Bernoulli distribution of parameter p_{ki}; by the conditional independence hypothesis, it results into

  $$p(\mathbf{x}|C_k) = \prod_{i=1}^d p_{ki}^{x_i}(1 - p_{ki})^{1-x_i}$$

  where $p_{ki} = p(x_i = 1|C_k)$ could be estimated by ML, as in the case of language models
- Functions $a_k(\mathbf{x})$ can then be defined as:

  $$a_k(\mathbf{x}) = \log(p(\mathbf{x}|C_k)p(C_k)) = \sum_{i=1}^D (x_i \log p_{ki} + (1 - x_i)\log(1 - p_{ki})) + \log p(C_k)$$

  These are still linear functions on $\mathbf{x}$.

- The same considerations can be done in the case of non binary features, where, for any class C_k, we may assume the value of the i-th feature is sampled from a distribution on a suitable domain (e.g. Poisson in the case of count data)

Generative models and the exponential family

The property that $p(C_k|\mathbf{x})$ is a generalized linear model with sigmoid (for the binary case) and softmax (for the multiclass case) activation function holds more in general than assuming a gaussian or bernoulli class conditional distribution $p(\mathbf{x}|C_k)$.

Generative models and the exponential family

Indeed, let the class conditional probability wrt C_k belong to the exponential family, that is it may be written in the general form

$$p(\mathbf{x}|C_k) = \frac{1}{s}g(\boldsymbol{\theta}_k)f\left(\frac{\mathbf{x}}{s}\right)e^{\frac{1}{s}\boldsymbol{\theta}_k^T\mathbf{u}(\mathbf{x})} = \exp\left(\frac{1}{s}\left(\boldsymbol{\theta}_k^T\mathbf{u}(\mathbf{x}) + A(\boldsymbol{\theta}_k, s)\right) + C\left(\frac{\mathbf{x}}{s}\right)\right)$$

Here,

1. $\boldsymbol{\theta}_k = (\theta_{k1}, \ldots, \theta_{km})$ is an m-dimensional array (for a give, suitable, m) denoted as the *natural parameter*
2. $\mathbf{u}$ is a function mapping $\mathbf{x}$ to an m-dimensional array $\mathbf{u}(\mathbf{x}) = (\mathbf{u}(\mathbf{x})_1, \ldots, \mathbf{u}(\mathbf{x})_m)$
3. s is a *dispersion* parameter
4. $g(\boldsymbol{\theta}_k)$ normalizes the function values so that $\int p(\mathbf{x}|C_k)d\mathbf{x} = 1$, hence $g(\boldsymbol{\theta}_k) = \frac{s}{\int f\left(\frac{\mathbf{x}}{s}\right)e^{\frac{1}{s}\boldsymbol{\theta}_k^T\mathbf{u}(\mathbf{x})d\mathbf{x}}}$; its inverse $\frac{s}{g(\boldsymbol{\theta}_k)}$ is denoted as the *partition function*
5. clearly, $A(\boldsymbol{\theta}_k, s) = \log\frac{g(\boldsymbol{\theta}_k)}{s}$ and $C\left(\frac{\mathbf{x}}{s}\right) = \log f\left(\frac{\mathbf{x}}{s}\right)$

Exponential family

Let us consider the gaussian distribution. The distribution belongs to the exponential family since

$$\begin{aligned}
p(x|\mu,\sigma) &= \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \\
&= \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} - \log\left(\sqrt{2\pi}\sigma\right)\right) \\
&= \exp\left(-\frac{x^2}{2\sigma^2} + x\frac{\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2} - \frac{1}{2}\log\left(2\pi\sigma^2\right)\right)
\end{aligned}$$

which fits the exponential family structure assuming $\boldsymbol{\theta} = (\frac{\mu}{\sigma^2}, -\frac{1}{\sigma^2})$, $\mathbf{u}(x) = (x, \frac{x^2}{2})$, $s = 1$, $A(\boldsymbol{\theta}, s) = -\frac{\mu^2}{2\sigma^2} - \log\sigma$, $C\left(\frac{\mathbf{x}}{s}\right) = -\frac{1}{2}\log(2\pi)$

Exponential family

Let us consider the bernoulli distribution $p(x|\pi) = \pi^x(1-\pi)^{1-x}$. The distribution belongs to the exponential family since

$$\begin{aligned}
p(x|\pi) &= \pi^x(1-\pi)^{1-x} \\
&= \exp\left(x\log\pi + (1-x)\log(1-\pi)\right) = \exp\left(x\log\frac{\pi}{1-\pi} + \log(1-\pi)\right)
\end{aligned}$$

which fits the exponential family structure assuming $\theta = \log\frac{\pi}{1-\pi}$, $u(x) = x$, $s = 1$, $A(\theta, s) = \log(1-\pi)$, $C\left(\frac{\mathbf{x}}{s}\right) = 0$

Generative models and the exponential family

In the case of binary classification, we check that $a(\mathbf{x})$ is a linear function

$$\begin{aligned}
a(\mathbf{x}) &= \log\frac{p(\mathbf{x}|\boldsymbol{\theta}_1)p(\boldsymbol{\theta}_1)}{p(\mathbf{x}|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)} = \log\frac{g(\boldsymbol{\theta}_1)e^{\frac{1}{s}\boldsymbol{\theta}_1^T\mathbf{u}(\mathbf{x})}p(\boldsymbol{\theta}_1)}{g(\boldsymbol{\theta}_2)e^{\frac{1}{s}\boldsymbol{\theta}_2^T\mathbf{u}(\mathbf{x})}p(\boldsymbol{\theta}_2)} \\
&= (\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2)^T\mathbf{x} + \log g(\boldsymbol{\theta}_1) - \log g(\boldsymbol{\theta}_2) + \log p(\boldsymbol{\theta}_1) - \log p(\boldsymbol{\theta}_2)
\end{aligned}$$

Similarly, for multiclass classification, we may easily derive that

$$a_k(\mathbf{x}) = \boldsymbol{\theta}_k^T \mathbf{x} + \log g(\boldsymbol{\theta}_k) + p(\boldsymbol{\theta}_k)$$

for all k.