Machine learning

Probabilistic classification - generative models

Corso di Laurea Magistrale in Informatica

Università di Roma Tor Vergata

Prof. Giorgio Gambosi

a.a. 2021-2022

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

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

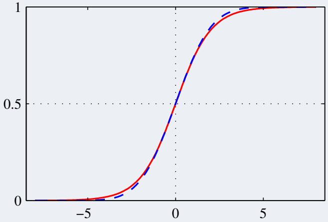
  that is, a is the log of the ratio between the posterior probabilities (log odds)
- We obtain that

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

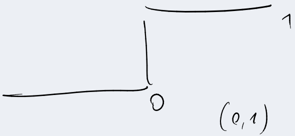
- $\sigma(x)$ is the logistic function or (sigmoid)

Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



Versione "smooth" della soglia:



Useful properties of the sigmoid

- $\sigma(-x) = 1 - \sigma(x)$ $\longrightarrow$ deriva dal fatto che e' rovesciato fra positivi e negativi
- $\dfrac{d\sigma(x)}{dx} = \sigma(x)(1-\sigma(x))$

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 $k = 1, \ldots, K$

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

- 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 $j \neq k$

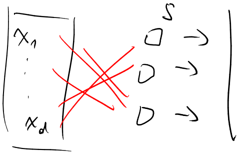
Usiamo, per stimare $p(C_k|\mathbf{x})$:

- caso binario: usiamo la sigmoide $p(C_1|\mathbf{x}) = \sigma(w^T\mathbf{x})$
- $K>2$: $p(C_i|\mathbf{x}) = s(w^T_i \mathbf{x})$

E' una funzione che $\to 1$ se un valore è più grande degli altri.

X feature x_1
$\vdots$
x_d

e le classi sono K: è come avere K elementi, ognuno dei quali fa una combinazione lineare delle feature, a cui viene applicata la S.



escono fuori valori di probabilità. È così che è fatto l'ultimo layer di una rete neurale: i primi n-1 layer cambiano la rappresentazione dei dati, sulla finale si fa softmax classification. Questo cambiamento è appreso dai dati; la rete individua una buona rappresentazione dei dati per ben classificare con softmax (o logistic regression per $k=2$).

Prima: $p(C_k|\mathbf{x}) = \sigma(w^T x)$ che e' l'ip parametrica, poi cerco i migliori w^T. Qui considero $p(\mathbf{x}|C_k)$ e definisco una struttura della distribuzione, es. Gaussiana D-variata (per D feature)

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,

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

I parametri migliori si trovano cercando $\forall$ classe, la migliore Gaussiana per poi invertire con Bayes.

Se tutte le Gaussiane fossero indipendenti, doveri cercare le diverse medie e matrici di covarianza che devo imparare, quindi parametri tutti diversi e modello più complesso.
Oppure dire che hanno tutte la stessa matrice di covarianza diversa, o ancora semplificare ulteriormente e dire che tutte le matrici di covarianza sono diagonali. O ancora, dire che la matrice di covarianza è uguale, diagonale e con tutti valori uguali:

$$\begin{pmatrix} \lambda & \dots & 0 \\ 0 & \dots & \lambda \end{pmatrix}$$ (matrice diagonale)

Noi assumiamo che abbiano tutte la stessa matrice di covarianza , così da ottenere il $p(x|C_k)$ che vediamo sopra

If $K = 2$,

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

where

(μ_1, Σ)

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

(μ_2, Σ)

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

$$= \frac{1}{2}(\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 - \mathbf{x}^T\Sigma^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_2^T\Sigma^{-1}\mathbf{x}) -$$

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

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

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

cost. rispetto ad x

cost. rispetto ad x (e' il prior)

Then,

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

with

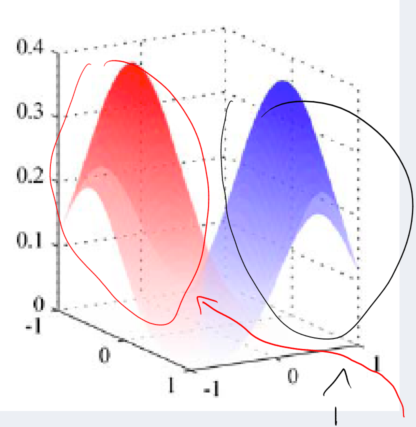
$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

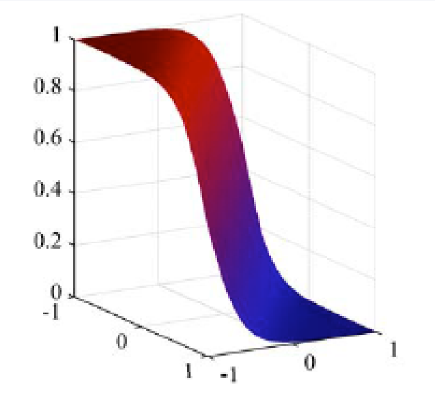
deriva dai parametri appresi per rappresentare la distribuzione

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

$p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$ 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

$$\mathbf{w}^T\mathbf{x} + w_0 = 0 \quad \longleftarrow \quad \text{iperpiano di separazione.}$$

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

Anche a più classi, abbiamo che la stima effettuata $\forall$ classe k è:

$s(w^T_k x + w_{0k})$

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,

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

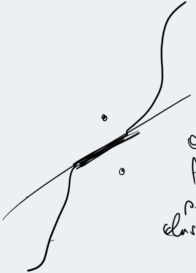
hence

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

that is,

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

As shown, this implies that boundaries are linear.



derivano dal fatto che le prob. delle due classi sono uguali.

General covariance matrices, binary case

Caso generale.

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

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

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

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

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

È la differenza di due funzioni quadratiche uguali, in un caso definita da μ_1 e nell'altro da μ_2.

è come avere (in una dimensione) $x\,k\,x$

General covariance, multiple classe

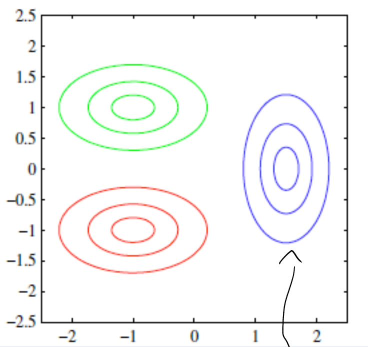
It can be proved that boundary surfaces are at most quadratic.

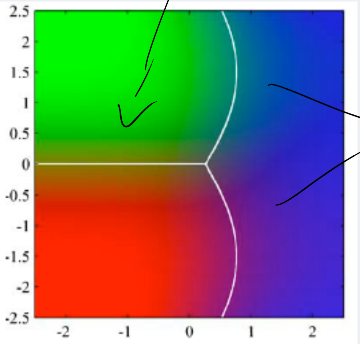
Example

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

Right: posterior distribution of classes, with boundary surfaces.

superficie di separazione lineare





superfici di separazione quadratiche.

matrice di cov. diverse.

$\mu_1, \mu_2, \Sigma_1, \Sigma_2$ stimate per max. 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 $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma$, and $\pi = p(C_1)$ (clearly, $p(C_2) = 1 - \pi$).

A questo punto ribaltiamo: ho $\mathcal{T}$ e cerco μ_1, μ_2, Σ e π che mi massimizzino la likelihood. Farò poi la log likelihood

Training set $\mathscr{T}$: 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 $p(\mathbf{x}, C_1) = p(\mathbf{x}|C_1)p(C_1) = \pi \cdot N(\mathbf{x}|\boldsymbol{\mu}_1, \Sigma)$

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

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

Genero n coppie (elem, target) fissati i param.

$$L(\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma|\mathscr{T}) = \prod_{i=1}^{n} \underbrace{(\pi \cdot N(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma))}_{\text{se } t_i=1 \text{ ho solo questo}}{}^{t_i} \underbrace{((1-\pi) \cdot N(\mathbf{x}_i|\boldsymbol{\mu}_2, \Sigma))}_{\text{se } t_i=0 \text{ ho solo questo}}{}^{1-t_i}$$

Estraggo n elementi a caso da Gauss. con parametri μ_1 μ_2 Σ

GDA and maximum likelihood

The corresponding log likelihood is

$$l(\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma|\mathscr{T}) = \sum_{i=1}^{n} \left(t_i \log \pi + t_i \log(N(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma))\right) +$$
$$+ \sum_{i=1}^{n} \left((1-t_i)\log(1-\pi) + (1-t_i)\log(N(\mathbf{x}_i|\boldsymbol{\mu}_2, \Sigma))\right)$$

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

$$\begin{pmatrix} \frac{\partial}{\partial \mu_{1,1}} \\ \frac{\partial}{\partial \mu_{1,2}} \\ \vdots \\ \frac{\partial}{\partial \mu_{1,d}} \end{pmatrix}$$

e' il vettore gradiente

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

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

Avrei messa matrice di derivate per via dei vari coefficienti.

Maximizing the log-likelihood wrt Σ provides

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

where

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

matrice di correlazione derivata dall'osservazione della I classe.

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

stesso per la II classe

and let

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

w

$\mu_1, \mu_2, \Sigma, \hat{\pi}$

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

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

Ricordiamo nel caso dei documenti.

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)

Vale sempre che ottengo un modello lineare generalizzato? Sì, in molti casi ma non sempre.

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 $p(\mathbf{x}|C_k)$.

Vale per una ampia famiglia di distribuzioni di probabilità.

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 $C\left(\frac{\mathbf{x}}{s}\right) = \log f\left(\frac{\mathbf{x}}{s}\right)$

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 $\boldsymbol{\theta} = (\frac{\mu}{\sigma^2}, -\frac{1}{\sigma^2})$, $\mathbf{u}(x) = (x, \frac{x^2}{2})$, $s = 1$, $A(\boldsymbol{\theta}, s) = -\frac{\mu^2}{2\sigma^2} - \log\sigma$, $C\left(\frac{\mathbf{x}}{s}\right) = -\frac{1}{2}\log\left(2\pi\right)$

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 $\theta = \log\frac{\pi}{1-\pi}$, $u(x) = x$, $s = 1$, $A(\theta, s) = \log(1-\pi)$, $C\left(\frac{\mathbf{x}}{s}\right) = 0$

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

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

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.

E' un approccio generale, se usato con funzioni $\in$ alla famiglia delle esponenziali