Probabilistic classification: generative models

Course of Machine Learning
Master Degree in Computer Science
University of Rome “Tor Vergata”

a.a. 2020-2021

Giorgio Gambosi
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).
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 $C_1$ of documents known to belong to $C_1$ and a similar collection $C_2$ 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 $C_1$ and $C_2$
Computing $p(C_k)$

The prior probabilities $p(C_k)$ ($k = 1, 2$) can be easily estimated from $C_1, C_2$: for example, by applying ML, we obtain

$$p(C_k) = \frac{|C_1|}{|C_1| + |C_2|}$$
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

$$p(d|C_k) = p(\bar{t}_1, \ldots, \bar{t}_{n_d}|C_k)$$
$$= p(\bar{t}_1|C_k)p(\bar{t}_2|\bar{t}_1, C_k) \cdots p(\bar{t}_{n_d}|\bar{t}_1, \ldots, \bar{t}_{n_d-1}, C_k)$$
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 \( k = 1, 2 \),

\[
p(\tilde{t}_i, \tilde{t}_j | C_k) = p(\tilde{t}_i | C_k)p(\tilde{t}_j | C_k)
\]
as, a consequence,

\[
p(d|C_k) = \prod_{j=1}^{n_d} p(\tilde{t}_j | C_k)
\]

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 \( p(\tilde{t}_j | C_k) \) have been derived for \( C_1 \) and \( C_2 \), respectively from documents in \( C_1 \) and \( C_2 \).
Generative models

- Classes are modeled by suitable conditional distributions $p(x|C_k)$ (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 $p(C_k|x)$ given such models (and the prior distributions $p(C_k)$ of classes)
- We may derive the parameters of $p(x|C_k)$ and $p(C_k)$ from the dataset, for example through maximum likelihood estimation
- Classification is performed by comparing $p(C_k|x)$ for all classes
Deriving posterior probabilities

- Let us consider the binary classification case and observe that

\[
p(C_1|x) = \frac{p(x|C_1)p(C_1)}{p(x|C_1)p(C_1) + p(x|C_2)p(C_2)} = \frac{1}{1 + \frac{p(x|C_2)p(C_2)}{p(x|C_1)p(C_1)}}
\]

- Let us define

\[
a = \log \frac{p(x|C_1)p(C_1)}{p(x|C_2)p(C_2)} = \log \frac{p(C_1|x)}{p(C_2|x)}
\]

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

- We obtain that

\[
p(C_1|x) = \frac{1}{1 + e^{-a}} = \sigma(a) \quad p(C_2|x) = 1 - \frac{1}{1 + e^{-a}} = \frac{1}{1 + e^a}
\]

- \( \sigma(x) \) is the logistic function or (sigmoid)
Useful properties of the sigmoid

- \( \sigma(-x) = 1 - \sigma(x) \)
- \( \frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x)) \)
Deriving posterior probabilities

- In the case \( K > 2 \), the general formula holds:

\[
p(C_k|x) = \frac{p(x|C_k)p(C_k)}{\sum_j p(x|C_j)p(C_j)}
\]

- Let us define, for each \( k = 1, \ldots, K \):

\[
a_k(x) = \log(p(x|C_k)p(C_k)) = \log p(C_k|x) + \log p(C_k)
\]

- Then, we may write:

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

- \( s(x) \) is the softmax function (or normalized exponential) and it can be seen as an extension of the sigmoid to the case \( K > 2 \).

- \( s(x) \) can be seen as a smoothed version of the maximum:

  if \( a_k \gg a_j \) for all \( j \neq k \), then \( s(a_k) \approx 1 \) and \( s(a_j) \approx 0 \) for all \( j \neq k \).
In Gaussian discriminant analysis (GDA) all class conditional distributions $p(x|C_k)$ are assumed gaussians. This implies that the corresponding posterior distributions $p(C_k|x)$ can be easily derived.

**Hypothesis**

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

$$p(x|C_k) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_k)^T \Sigma^{-1}(x - \mu_k)\right)$$
If $K = 2$,

$$p(C_1 | x) = \sigma(a(x))$$

where

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

$$= \log \frac{\frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (x - \mu_1)^T \Sigma^{-1} (x - \mu_1) \right) p(C_1)}{\frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (x - \mu_2)^T \Sigma^{-1} (x - \mu_2) \right) p(C_2)}$$

$$= \frac{1}{2} (\mu_2^T \Sigma^{-1} \mu_2 - x^T \Sigma^{-1} \mu_2 - \mu_2^T \Sigma^{-1} x) - \frac{1}{2} (\mu_1^T \Sigma^{-1} \mu_1 - x^T \Sigma^{-1} \mu_1 - \mu_1^T \Sigma^{-1} x) + \log \frac{p(C_1)}{p(C_2)}$$
Binary case

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

$$x^T \Sigma^{-1} \mu_1 = \mu_1^T \Sigma^{-1} x$$
$$x^T \Sigma^{-1} \mu_2 = \mu_2^T \Sigma^{-1} x$$

Then,

$$a(x) = \frac{1}{2} (\mu_2^T \Sigma^{-1} \mu_2 - \mu_1^T \Sigma^{-1} \mu_1) + (\mu_1^T \Sigma^{-1} - \mu_2^T \Sigma^{-1}) x + \log \frac{p(C_1)}{p(C_2)} = w^T x + w_0$$

with

$$w = \Sigma^{-1} (\mu_1 - \mu_2)$$
$$w_0 = \frac{1}{2} (\mu_2^T \Sigma^{-1} \mu_2 - \mu_1^T \Sigma^{-1} \mu_1) + \log \frac{p(C_1)}{p(C_2)}$$

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

Example

Left, the class conditional distributions $p(x|C_1), p(x|C_2)$, gaussians with $D = 2$. Right the posterior distribution of $C_1$, $p(C_1|x)$ with sigmoidal slope.
The discriminant function can be obtained by the condition \( p(C_1|x) = p(C_2|x) \), that is, \( \sigma(a(x)) = \sigma(-a(x)) \).

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

\[
\mathbf{w}^T \mathbf{x} + w_0 = 0
\]
or

\[
\Sigma^{-1} (\mu_1 - \mu_2) \mathbf{x} + 1/2 (\mu_2^T \Sigma^{-1} \mu_2 - \mu_1^T \Sigma^{-1} \mu_1) + \log p(C_2) - p(C_1) = 0
\]

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

\[
2(\mu_2 - \mu_1) \mathbf{x} + \| \mu_1 \|^2 - \| \mu_2 \|^2 + 2\lambda \log \frac{p(C_2)}{p(C_1)} = 0
\]
In this case, we refer to the softmax function:

$$p(C_k|x) = s(a_k(x))$$

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

By the above considerations, it easily turns out that

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

Again, $$p(C_k|x) = \sigma(\mathbf{w}_k^T \mathbf{x} + w_0)$$ 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,

\[
p(C_k|\mathbf{x}) = p(C_j|\mathbf{x}) \quad \text{and} \quad p(C_i|\mathbf{x}) < p(C_k|\mathbf{x}) \quad i \neq j, k
\]

hence

\[
e^{a_k(\mathbf{x})} = e^{a_j(\mathbf{x})} \quad \text{and} \quad e^{a_i(\mathbf{x})} < e^{a_k(\mathbf{x})} \quad i \neq j, k
\]

that is,

\[
a_k(\mathbf{x}) = a_j(\mathbf{x}) \quad \text{and} \quad a_i(\mathbf{x}) < a_k(\mathbf{x}) \quad i \neq j, k
\]

As shown, this implies that boundaries are linear.
General covariance matrices, binary case

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

\[
a(x) = \log \frac{p(x|C_1)p(C_1)}{p(x|C_2)p(C_2)} \\
= \log \frac{\exp\left(-\frac{1}{2}(x - \mu_1)^T \Sigma_1^{-1}(x - \mu_1)\right)}{\exp\left(-\frac{1}{2}(x - \mu_2)^T \Sigma_2^{-1}(x - \mu_2)\right)} + \frac{1}{2} \log \frac{|\Sigma_2|}{|\Sigma_1|} + \log \frac{p(C_1)}{p(C_2)} \\
= \frac{1}{2} \left((x - \mu_2)^T \Sigma_2^{-1}(x - \mu_2) - (x - \mu_1)^T \Sigma_1^{-1}(x - \mu_1)\right) + \frac{1}{2} \log \frac{|\Sigma_2|}{|\Sigma_1|} + \log \frac{p(C_1)}{p(C_2)}
\]
By applying the same considerations, the decision boundary turns out to be

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

Classes are separated by a (at most) quadratic surface.
General covariance, multiple classes

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 $p(x|C_k)$ 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 $\Sigma$.

It is then necessary to estimate $\mu_1, \mu_2, \Sigma$, and $\pi = p(C_1)$ (clearly, $p(C_2) = 1 - \pi$).
Training set $\mathcal{T}$: includes $n$ elements $(x_i, t_i)$, with

$$t_i = \begin{cases} 
0 & \text{if } x_i \in C_2 \\
1 & \text{if } x_i \in C_1 
\end{cases}$$

If $x \in C_1$, then $p(x, C_1) = p(x|C_1)p(C_1) = \pi \cdot \mathcal{N}(x|\mu_1, \Sigma)$

If $x \in C_2$, $p(x, C_2) = p(x|C_2)p(C_2) = (1 - \pi) \cdot \mathcal{N}(x|\mu_2, \Sigma)$

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

$$L(\pi, \mu_1, \mu_2, \Sigma|\mathcal{T}) = \prod_{i=1}^{n} (\pi \cdot \mathcal{N}(x_i|\mu_1, \Sigma))^{t_i} ((1 - \pi) \cdot \mathcal{N}(x_i|\mu_2, \Sigma))^{1-t_i}$$
The corresponding log likelihood is

\[
l(\pi, \mu_1, \mu_2, \Sigma | T) = \sum_{i=1}^{n} (t_i \log \pi + t_i \log(\mathcal{N}(x_i | \mu_1, \Sigma))) + \sum_{i=1}^{n} ((1 - t_i) \log(1 - \pi) + (1 - t_i) \log(\mathcal{N}(x_i | \mu_2, \Sigma)))
\]

Its derivative wrt \( \pi \) 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

\[
\pi = \frac{n_1}{n}
\]
The maximum wrt $\mu_1$ (and $\mu_2$) is obtained by computing the gradient

$$\frac{\partial l}{\partial \mu_1} = \frac{\partial}{\partial \mu_1} \sum_{i=1}^{n} t_i \log(\mathcal{N}(x_i|\mu_1, \Sigma)) = \cdots = \Sigma^{-1} \sum_{i=1}^{n} t_i (x_i - \mu_1)$$

As a consequence, we have $\frac{\partial l}{\partial \mu_1} = 0$ for

$$\sum_{i=1}^{n} t_i x_i = \sum_{i=1}^{n} t_i \mu_1$$

hence, for

$$\mu_1 = \frac{1}{n_1} \sum_{x_i \in C_1} x_i$$
Similarly, \( \frac{\partial l}{\partial \mu_2} = 0 \) for

\[
\mu_2 = \frac{1}{n_2} \sum_{x_i \in C_2} x_i
\]
GDA and maximum likelihood

Maximizing the log-likelihood wrt $\Sigma$ provides

$$\Sigma = \frac{n_1}{n} S_1 + \frac{n_2}{n} S_2$$

where

$$S_1 = \frac{1}{n_1} \sum_{x_i \in C_1} (x_i - \mu_1)(x_i - \mu_1)^T$$

$$S_2 = \frac{1}{n_2} \sum_{x_i \in C_2} (x_i - \mu_2)(x_i - \mu_2)^T$$

and let

$$S = \frac{n_1}{n} S_1 + \frac{n_2}{n} S_2$$
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

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

where $p_{ki} = p(x_i = 1|C_k)$ could be estimated by ML, as in the case of language models.

Functions $a_k(x)$ can then be defined as:

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

These are still linear functions on $x$.

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 $p(C_k|x)$ 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 $p(x|C_k)$. 
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

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

Here,

1. $\theta_k = (\theta_{k1}, \ldots, \theta_{km})$ is an $m$-dimensional array (for a given, suitable, $m$) denoted as the natural parameter
2. $u$ is a function mapping $x$ to an $m$-dimensional array $u(x) = (u(x)_1, \ldots, u(x)_m)$
3. $s$ is a dispersion parameter
4. $g(\theta_k)$ normalizes the function values so that $\int p(x|C_k)dx = 1$, hence $g(\theta_k) = \frac{s}{\int f\left( \frac{x}{s} \right) e^{\frac{1}{s} \theta_k^T u(x)}dx}$; its inverse $s g(\theta_k)$ is denoted as the partition function
5. clearly, $A(\theta_k, s) = \log \frac{g(\theta_k)}{s}$ and $C \left( \frac{x}{s} \right) = \log f \left( \frac{x}{s} \right)$
Let us consider the gaussian distribution. The distribution belongs to the exponential family since

\[ p(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp \left( -\frac{(x - \mu)^2}{2\sigma^2} \right) \]

\[ = \exp \left( -\frac{(x - \mu)^2}{2\sigma^2} - \log \left( \sqrt{2\pi\sigma} \right) \right) \]

\[ = \exp \left( -\frac{x^2}{2\sigma^2} + x \frac{\mu}{\sigma^2} - \frac{\mu^2}{2\sigma^2} - \frac{1}{2} \log \left( 2\pi\sigma^2 \right) \right) \]

which fits the exponential family structure assuming \( \theta = (\frac{\mu}{\sigma^2}, -\frac{1}{\sigma^2}) \), \( u(x) = (x, \frac{x^2}{2}) \), \( s = 1 \), \( A(\theta, s) = -\frac{\mu^2}{2\sigma^2} - \log \sigma \), \( C \left( \frac{x}{s} \right) = -\frac{1}{2} \log (2\pi) \)
Let us consider the bernoulli distribution \( p(x|\pi) = \pi^x (1 - \pi)^{1-x} \). The distribution belongs to the exponential family since

\[
p(x|\pi) = \pi^x (1 - \pi)^{1-x} \\
= \exp (x \log \pi + (1 - x) \log(1 - \pi)) = \exp \left( x \log \frac{\pi}{1 - \pi} + \log(1 - \pi) \right)
\]

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