

# Probabilistic learning

Course of Machine Learning  
Master Degree in Computer Science  
University of Rome "Tor Vergata"  
a.a. 2023-2024

Giorgio Gambosi

## Supervised learning framework: deriving a probabilistic predictor

As done before, we assume that the observed dataset (features and target) has been derived by randomly sampling:

- $\mathcal{X}$  according to the probability distribution  $p_{\mathcal{D}_1}(x)$  (usually the uniform distribution)
- $\mathcal{Y}$  according to the conditional distribution  $p_{\mathcal{D}_2}(t|x)$

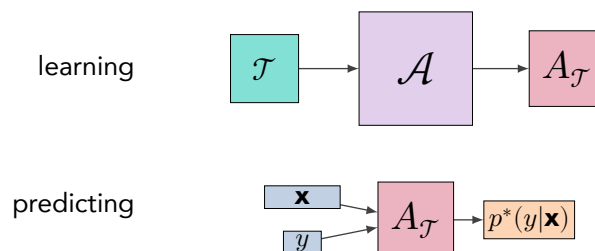
Deriving a probabilistic predictor results into deriving, from the training set  $\mathcal{T}$ , an algorithm computing a conditional distribution  $\hat{p}(t|\mathbf{x})$  which approximates the correct, unknown distribution  $\mathcal{D}_2$ .

An independent **decision strategy** must be applied to  $p^*(t|\mathbf{x})$  to return a specific prediction  $h(\mathbf{x})$

## Supervised learning framework: deriving a probabilistic predictor

First approach

1. we may then consider a class of possible conditional distributions  $\mathcal{P}$  and
2. select (infer) the "best" conditional distribution  $p^* \in \mathcal{P}$  from the available knowledge (that is, the dataset), according to some measure  $q$
3. given any new item  $x$ , apply  $p^*(t|\mathbf{x})$  to assign probabilities for each possible value of the corresponding target



## Inferring a best distribution

How to define the class of possible conditional distributions  $p(t|\mathbf{x})$ ?

- usually, parametric approach: distributions defined by a common (arbitrary) structure and a set of parameters

Example: logistic regression for binary classification

The probability  $p(t|\mathbf{x})$ , where  $t \in \{0, 1\}$ , is assumed to be a Bernoulli distribution

$$p(t|\mathbf{x}) = \pi(\mathbf{x})^t (1 - \pi(\mathbf{x}))^{1-t}$$

with

$$\pi(\mathbf{x}) = p(t = 1|\mathbf{x}) = \frac{1}{1 + e^{-\sum_{i=1}^d w_i x_i + w_0}}$$

### Inferring a best distribution

What is a measure  $q(p, \mathcal{T})$  of the quality of the distribution (given the dataset  $\mathcal{T} = (\mathbf{X}, \mathbf{t})$ )?

- this is related to how a dataset generated by randomly sampling from  $\mathcal{D}_1$  (usually uniform) and  $p(t|\mathbf{x})$  (instead of the unknown distribution  $\mathcal{D}_2$ ) could be similar to the available dataset  $\mathcal{T}$
- in particular, what is the probability that the dataset  $\mathcal{T} = (\mathbf{X}, \mathbf{t})$  is obtained under the following hypotheses?
  - $n = |\mathbf{t}|$  pairs  $\mathbf{x}_i, t_i$  are each other independently sampled
  - $\mathbf{x}_i$  is sampled from  $\mathcal{D}_1$  (which we assume uniform)
  - $t_i$  is sampled from  $p(t|\mathbf{x}_i)$
- we may use such probability as the quality measure  $q(p, \mathcal{T})$  and search the distribution  $p^*(t|\mathbf{x})$  that makes  $p(\mathbf{X}, \mathbf{t})$  maximum assuming  $\mathcal{D}_1$  is the uniform distribution and  $\mathcal{D}_2$  is  $p^*(t|\mathbf{x})$

### Inferring a best distribution

That is, we consider the probability

$$p(\mathbf{X}, \mathbf{t}) = \prod_{i=1}^n p(\mathbf{x}_i, t_i) = \prod_{i=1}^n p(t_i|\mathbf{x}_i)p(\mathbf{x}_i) \propto \prod_{i=1}^n p(t_i|\mathbf{x}_i) = q(p, \mathcal{T})$$

and look (within some class of distributions) for the conditional probability  $p^*(t|\mathbf{x})$  which makes  $p(\mathbf{X}, \mathbf{t})$  maximum

Observe that learning the distribution  $p^*(t|\mathbf{x})$  which maximizes  $q(p, \mathcal{T})$  corresponds, in the probabilistic predictor case, to learning the function  $h^*$  which minimizes the empirical risk  $\overline{\mathcal{R}}_{\mathcal{T}}(h)$  in the functional predictor case. In both cases, learning is performed through optimization.

### Inferring a best distribution

The same considerations done wrt the inductive bias in the case of a functional predictor, and related to overfitting and underfitting, can be rephrased here wrt the class of possible conditional distributions.

### A different approach

Instead of finding a best distribution  $p^* \in \mathcal{P}$  and use it to predict target probabilities as  $p^*(y|\mathbf{x})$  for any element  $\mathbf{x}$ , we could

- consider for each possible conditional distribution  $p \in \mathcal{P}$  its quality  $q(p, \mathcal{T})$

- compose all conditional distributions  $p(y|\mathbf{x})$  each weighted by its quality  $q(p, \mathcal{T})$  (for example by means of a weighted averaging)
- apply the resulting distribution

### Different strategies

Assume  $q$  takes the form of a probability distribution (of probability distribution)

- first approach: take the modal value (the distribution of maximum quality) and apply it to perform predictions
- second approach: compute the expectation of the distributions, wrt the probability distribution  $q$

### Inference of predictive distribution

#### Dataset

We assume elements in  $\mathcal{T}$  correspond to a set of  $n$  samples, independently drawn from the same probability distribution (that is, they are **independent and identically distributed**, i.i.d): they can be seen as  $n$  realizations of a single random variable.

We are interested in learning, starting from  $\mathcal{T}$ , a **predictive distribution**  $p(\mathbf{x}|\mathbf{X})$  (or  $p(\mathbf{x}, t|\mathbf{X}, \mathbf{t})$ ) for any new element (or element-target pair). We may interpret this as the probability that, in a random sampling, the element actually returned is indeed  $\mathbf{x}$  (or  $\mathbf{x}, t$ ).

- in the case that  $\mathcal{T} = \mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ , we are interested in deriving the probability distribution  $p(\mathbf{x}|\mathbf{X})$  of a new element, given the knowledge of the set  $\mathbf{X}$
- in the case that  $\mathcal{T} = (\mathbf{X}, \mathbf{t}) = \{(\mathbf{x}_1, t_1), \dots, (\mathbf{x}_n, t_n)\}$ , we are interested in deriving the joint probability distribution  $p(\mathbf{x}, t|\mathbf{X}, \mathbf{t})$  or, assuming  $p(\mathbf{x}|\mathbf{X}, \mathbf{t})$  uniform and thus also independent from  $\mathbf{X}, \mathbf{t}$ , the conditional distribution  $p(t|\mathbf{x}, \mathbf{X}, \mathbf{t})$ , given the knowledge of the set of pairs  $\mathbf{X}, \mathbf{t}$

### Probabilistic models

A **probabilistic model** is a collection of probability distributions with the same structure, defined over the data domain. Probability distribution are instances of the probabilistic model and are characterized by the values assumed by a set of **parameters**.

*Example 1.* In a bivariate gaussian probabilistic model, distributions are characterized by the values assumed by:

1. the mean  $\mu = (\mu_1, \mu_2)$
2. the covariance matrix  $\Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix}$

where  $\sigma_{12} = \sigma_{21}$

A probabilistic model could be

**Parametric** if the set of parameters is given, finite, and independent from the data

**Non parametric** if the set of parameters is not given in advance, but derives from the data

### Likelihood

Given a dataset  $\mathcal{T}$  and a probability distribution  $p$  of parameters  $\theta$  defined on the same data domain,

- the **likelihood** of  $\theta$  wrt  $\mathcal{T}$  is defined as

$$L(\theta|\mathcal{T}) = p(\mathcal{T}|\theta)$$

the probability of the dataset (that the dataset is generated) under distribution  $p$  with parameters  $\theta$

- while the probability  $p(\mathcal{T}|\theta)$  is considered as a function of  $p(\mathcal{T}|\theta)$  with  $\theta$  fixed, the likelihood  $L(\theta|\mathcal{T})$  is a function of  $\theta$  with  $\mathcal{T}$  fixed
- parameters  $\theta$  are considered as (independent) variables (**frequentist interpretation** of probability)

### Likelihood

- By assuming that elements in  $\mathcal{T}$  are i.i.d.,

$$L(\theta|\mathcal{T}) = p(\mathbf{X}|\theta) = \prod_{i=1}^n p(\mathbf{x}_i|\theta) \quad \text{in the first case}$$

$$\begin{aligned} L(\theta|\mathcal{T}) &= p(\mathbf{X}, \mathbf{t}|\theta) = \prod_{i=1}^n p(\mathbf{x}_i, t_i|\theta) = \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta)p(\mathbf{x}_i|\theta) = p(\mathbf{x}|\theta) \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta) \\ &= p(\mathbf{x}) \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta) \propto \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta) \quad \text{in the second case, assuming } p(\mathbf{x}|\theta) \text{ uniform} \end{aligned}$$

### Maximum likelihood estimate

#### Approach

**Frequentist** point of view: parameters are deterministic variables, whose value is unknown and must be estimated.

Determine the parameter value that maximize the likelihood

$$\theta^* = \underset{\theta}{\operatorname{argmax}} L(\theta|\mathcal{T}) = \underset{\theta}{\operatorname{argmax}} p(\mathbf{X}|\theta) = \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^n p(\mathbf{x}_i|\theta)$$

or

$$\theta^* = \underset{\theta}{\operatorname{argmax}} L(\theta|\mathcal{T}) = \underset{\theta}{\operatorname{argmax}} p(\mathbf{X}, \mathbf{t}|\theta) = \underset{\theta}{\operatorname{argmax}} p(\mathbf{x}) \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta) = \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^n p(t_i|\mathbf{x}_i, \theta)$$

### Maximum likelihood estimate

## Log-likelihood

$$l(\theta|\mathcal{T}) = \ln L(\theta|\mathcal{T})$$

is usually preferable, since products are turned into sums, while  $\theta^*$  remains the same (since log is a monotonic function), that is

$$\operatorname{argmax}_{\theta} l(\theta|\mathcal{T}) = \operatorname{argmax}_{\theta} L(\theta|\mathcal{T})$$

## Estimate

$$\theta_{ML}^* = \operatorname{argmax}_{\theta} p(\mathbf{X}|\theta) = \operatorname{argmax}_{\theta} \sum_{i=1}^n \ln p(\mathbf{x}_i|\theta)$$

or

$$\theta_{ML}^* = \operatorname{argmax}_{\theta} p(\mathbf{X}, \mathbf{t}|\theta) = \operatorname{argmax}_{\theta} \sum_{i=1}^n \ln p(t_i|\mathbf{x}_i, \theta)$$

## Maximum likelihood estimate

## Solution

Solve the system

$$\frac{\partial l(\theta|\mathcal{T})}{\partial \theta_i} = 0 \quad i = 1, \dots, d$$

more concisely,

$$\nabla_{\theta} l(\theta|\mathcal{T}) = \mathbf{0}$$

## Prediction

Probability of a new observation  $\mathbf{x}$ :

$$p(\mathbf{x}|\mathbf{X}) = \int_{\theta} p(\mathbf{x}|\theta) p(\theta|\mathbf{X}) d\theta \approx \int_{\theta} p(\mathbf{x}|\theta_{ML}^*) p(\theta|\mathbf{X}) d\theta = p(\mathbf{x}|\theta_{ML}^*) \int_{\theta} p(\theta|\mathbf{X}) d\theta = p(\mathbf{x}|\theta_{ML}^*)$$

Predictive distribution  $t|\mathbf{x}$ :

$$p(t|\mathbf{x}, \mathbf{X}, \mathbf{t}) = \int_{\theta} p(t|\mathbf{x}, \theta) p(\theta|\mathbf{X}, \mathbf{t}) d\theta \approx \int_{\theta} p(t|\mathbf{x}, \theta_{ML}^*) p(\theta|\mathbf{X}) d\theta = p(\mathbf{x}|\theta_{ML}^*) \int_{\theta} p(\theta|\mathbf{X}, \mathbf{t}) d\theta = p(t|\mathbf{x}, \theta_{ML}^*)$$

## Maximum likelihood estimate

Example 2. Collection  $\mathbf{X}$  of  $n$  binary events, modeled through a Bernoulli distribution with unknown parameter  $\phi$

$$p(x|\phi) = \phi^x (1 - \phi)^{1-x}$$

$$\text{Likelihood: } L(\phi|\mathbf{X}) = \prod_{i=1}^n \phi^{x_i} (1 - \phi)^{1-x_i}$$

$$\text{Log-likelihood: } l(\phi|\mathbf{X}) = \sum_{i=1}^n (x_i \ln \phi + (1 - x_i) \ln(1 - \phi)) = n_1 \ln \phi + n_0 \ln(1 - \phi)$$

where  $n_0$  ( $n_1$ ) is the number of events  $x \in \mathbf{X}$  equal to 0 (1)

$$\frac{\partial l(\phi|\mathbf{X})}{\partial \phi} = \frac{n_1}{\phi} - \frac{n_0}{1 - \phi} = 0 \quad \Rightarrow \quad \phi^*_{ML} = \frac{n_1}{n_0 + n_1} = \frac{n_1}{n}$$

### Maximum likelihood estimate

Example 3. Linear regression: collection  $\mathbf{X}, \mathbf{t}$  of value-target pairs, modeled as  $p(\mathbf{x}, t) = p(\mathbf{x})p(t|\mathbf{x}, \mathbf{w}, \sigma^2)$ , with  $\mathbf{w} \in \mathbf{R}^d$ ,  $w_0 \in \mathbf{R}$ :

- $p(\mathbf{x})$  uniform
- $p(t|\mathbf{x}, \mathbf{w}, \sigma^2) = \mathcal{N}(\mathbf{w}^T \mathbf{x} + w_0, 1/\beta)$  ( $\beta$ , the inverse of the variance, is the **precision**)

$$\text{Likelihood: } L(\mathbf{t}|\mathbf{X}, \mathbf{w}, w_0, \beta) = \prod_{i=1}^n p(t_i|\mathbf{x}_i, \mathbf{w}, w_0, \beta) = \prod_{i=1}^n \mathcal{N}(\mathbf{w}^T \mathbf{x}_i + w_0, \beta)$$

Log-likelihood:

$$\begin{aligned} l(\mathbf{t}|\mathbf{X}, \mathbf{w}, w_0, \beta) &= \sum_{i=1}^n \ln p(t_i|\mathbf{x}_i, \mathbf{w}, w_0, \beta) = \sum_{i=1}^n \ln \left( \sqrt{\frac{\beta}{2\pi}} e^{-\frac{\beta(\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2}{2}} \right) = \sum_{i=1}^n \left( -\frac{\beta(\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2}{2} + \frac{1}{2} \ln \beta - \frac{1}{2} \ln 2\pi \right) \\ &= -\frac{\beta}{2} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2 + \frac{n}{2} \ln \beta - \frac{n}{2} \ln(2\pi) \end{aligned}$$

### Maximum likelihood estimate

Example 4.

$$\frac{\partial}{\partial w_k} l(\mathbf{t}|\mathbf{X}, \mathbf{w}, w_0, \beta) = -\frac{\beta}{2} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i) x_{ik} \quad k = 1, \dots, d$$

$$\frac{\partial}{\partial w_0} l(\mathbf{t}|\mathbf{X}, \mathbf{w}, w_0, \beta) = -\frac{\beta}{2} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)$$

$$\frac{\partial}{\partial \beta} l(\mathbf{t}|\mathbf{X}, \mathbf{w}, w_0, \beta) = -\frac{1}{2} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2 + \frac{n}{2\beta}$$

The ML estimation for  $\mathbf{w}, w_0$  (linear regression coefficients) is obtained as the solution of the  $(d+1, d+1)$  linear system

$$\begin{aligned} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i) x_{ik} &= 0 \quad k = 1, \dots, d \\ \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i) &= 0 \end{aligned}$$

The ML estimation for  $\beta$  is obtained by

$$-\frac{1}{2} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2 + \frac{n}{2\beta} = 0 \quad \Rightarrow \quad \beta_{ML} = \left( \frac{1}{n} \sum_{i=1}^n (\mathbf{w}^T \mathbf{x}_i + w_0 - t_i)^2 \right)^{-1}$$

## ML and overfitting

### Overfitting

Maximizing the likelihood of the observed dataset tends to result into an estimate too sensitive to the dataset values, hence into **overfitting**. The obtained estimates are suitable to model observed data, but may be too specialized to be used to model different datasets.

### Penalty functions

An additional function  $P(\theta)$  can be introduced with the aim to limit overfitting and the overall complexity of the model. This results in the following function to maximize

$$C(\theta|\mathbf{X}) = l(\theta|\mathbf{X}) - P(\theta)$$

as a common case,  $P(\theta) = \frac{\gamma}{2} \|\theta\|^2$ , with  $\gamma$  a **tuning** parameter.

## Maximum a posteriori estimate

### Idea

Inference through maximum a posteriori (MAP) is similar to ML, but  $\theta$  is now considered as a random variable (bayesian approach), whose distribution has to be derived from observations, also taking into account previous knowledge (prior distribution). The parameter value maximizing

$$p(\theta|\mathcal{T}) = \frac{p(\mathcal{T}|\theta)p(\theta)}{p(\mathcal{T})}$$

is computed.

## Maximum a posteriori estimate

### Estimate

$$\begin{aligned} \theta_{MAP}^* &= \underset{\theta}{\operatorname{argmax}} p(\theta|\mathcal{T}) = \underset{\theta}{\operatorname{argmax}} p(\mathcal{T}|\theta)p(\theta) \\ &= \underset{\theta}{\operatorname{argmax}} L(\theta|\mathcal{T})p(\theta) = \underset{\theta}{\operatorname{argmax}} (l(\theta|\mathcal{T}) + \ln p(\theta)) \end{aligned}$$

which results into

$$\theta_{MAP}^* = \underset{\theta}{\operatorname{argmax}} \left( \sum_{i=1}^n \ln p(\mathbf{x}_i|\theta) + \ln p(\theta) \right)$$

or

$$\theta_{MAP}^* = \underset{\theta}{\operatorname{argmax}} \left( \sum_{i=1}^n \ln p(t_i | \mathbf{x}_i, \theta) + \ln p(\theta) \right)$$

## MAP and gaussian prior

### Hypothesis

Assume  $\theta$  is distributed around the origin as a multivariate gaussian with uniform variance and null covariance. That is,

$$p(\theta) \sim \mathcal{N}(\theta | \mathbf{0}, \sigma^2) = \frac{1}{(2\pi)^{d/2} \sigma^d} e^{-\frac{\|\theta\|^2}{2\sigma^2}} \propto e^{-\frac{\|\theta\|^2}{2\sigma^2}}$$

### Inference

From the hypothesis,

$$\begin{aligned} \theta_{MAP}^* &= \underset{\theta}{\operatorname{argmax}} p(\theta | \mathcal{T}) = \underset{\theta}{\operatorname{argmax}} (l(\theta | \mathcal{T}) + \ln p(\theta)) \\ &= \underset{\theta}{\operatorname{argmax}} \left( l(\theta | \mathcal{T}) + \ln e^{-\frac{\|\theta\|^2}{2\sigma^2}} \right) = \underset{\theta}{\operatorname{argmax}} \left( l(\theta | \mathcal{T}) - \frac{\|\theta\|^2}{2\sigma^2} \right) \end{aligned}$$

which is equal to the penalty function introduced before, if  $\gamma = \frac{1}{\sigma^2}$

## MAP estimate

*Example 5.* Collection  $\mathbf{X}$  of  $n$  binary events, modeled as a Bernoulli distribution with unknown parameter  $\phi$ . Initial knowledge of  $\phi$  is modeled as a Beta distribution:

$$p(\phi | \alpha, \beta) = \text{Beta}(\phi | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \phi^{\alpha-1} (1 - \phi)^{\beta-1}$$

Log-likelihood

$$\begin{aligned} l(\phi | \mathbf{X}) &= \sum_{i=1}^n (x_i \ln \phi + (1 - x_i) \ln(1 - \phi)) = n_1 \ln \phi + n_0 \ln(1 - \phi) \\ \frac{\partial}{\partial \phi} (l(\phi | \mathbf{X}) + \ln \text{Beta}(\phi | \alpha, \beta)) &= \frac{n_1}{\phi} - \frac{n_0}{1 - \phi} + \frac{\alpha - 1}{\phi} - \frac{\beta - 1}{1 - \phi} = 0 \quad \Rightarrow \\ \phi_{MAP}^* &= \frac{N_1 + \alpha - 1}{n_0 + n_1 + \alpha + \beta - 2} = \frac{n_1 + \alpha - 1}{n + \alpha + \beta - 2} \end{aligned}$$

## Note

### Gamma function



The function

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt$$

is an extension of the factorial to the real numbers field: in fact, for any integer  $x$ ,

$$\Gamma(x) = (x-1)!$$

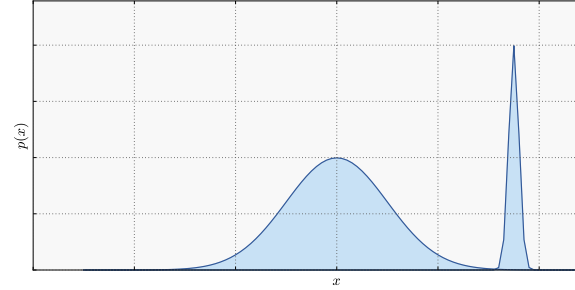
## Applying bayesian inference

### Mode and mean

Once the posterior distribution

$$p(\theta|\mathbf{X}) = \frac{p(\mathbf{X}|\theta)p(\theta)}{p(\mathbf{X})} = \frac{p(\mathbf{X}|\theta)p(\theta)}{\int_{\theta} p(\mathbf{X}|\theta)d\theta}$$

is available, MAP estimate computes the most probable value (mode)  $\theta_{MAP}$  of the distribution. This may lead



to inaccurate estimates, as in the figure below:

## Applying bayesian inference

### Mode and mean

A better estimation can be obtained by applying a fully bayesian approach and referring to the whole posterior distribution, for example by deriving the expectation of  $\theta$  w.r.t.  $p(\theta|\mathbf{X})$ ,

$$\theta^* = E_{p(\theta|\mathbf{X})}[\theta] = \int_{\theta} \theta p(\theta|\mathbf{X}) d\theta$$

## Bayesian estimate

*Example 6.* Collection  $\mathbf{X}$  of  $n$  binary events, modeled as a Bernoulli distribution with unknown parameter  $\phi$ . Initial knowledge of  $\phi$  is modeled as a Beta distribution:

$$p(\phi|\alpha, \beta) = \text{Beta}(\phi|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \phi^{\alpha-1} (1 - \phi)^{\beta-1}$$

Posterior distribution

$$\begin{aligned} p(\phi|\mathbf{X}, \alpha, \beta) &= \frac{\prod_{i=1}^N \phi^{x_i} (1 - \phi)^{1-x_i} p(\phi|\alpha, \beta)}{p(\mathbf{X})} \\ &= \frac{\phi^{N_1} (1 - \phi)^{N_0} \phi^{\alpha-1} (1 - \phi)^{\beta-1}}{\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} p(\mathbf{X})} = \frac{\phi^{N_1+\alpha-1} (1 - \phi)^{N_0+\beta-1}}{Z} \end{aligned}$$

Hence,

$$p(\phi|\mathbf{X}, \alpha, \beta) = \text{Beta}(\phi|\alpha + N_1, \beta + N_0)$$

## 1 Model selection

### Model selection

In the process described, a model (structure, hyper-parameter values) must be identified, in some way. How can we deal with this problem?

This is performed through **model selection**: identify, in a set of possible models, the one which we expect is best to represent the available data.

Indeed, the one whose best (or a good) instantiation is best to represent the available data

We need a way to compare models (not their instantiations), given the dataset

### Model selection in practice

### Validation

**Test set** Dataset is split into Training set (used for learning parameters) and Test set (used for measuring effectiveness). Good for large datasets: otherwise, small resulting training and test set (few data for fitting and validation)

**Cross validation** Dataset partitioned into  $K$  equal-sized sets. Iteratively, in  $K$  phases, use one set as test set and the union of the other  $K - 1$  ones as training set ( $K$ -fold cross validation). Average validation measures.

As a particular case, iteratively leave one element out and use all other points as training set (Leave-one-out cross validation).

Time consuming for large datasets and for models which are costly to fit.

### Model selection in practice

### Information measures

Faster methods to compare model effectiveness, based on computing measures which take into account data fitting and model complexity.

**Akaike Information Criterion (AIC)** Let  $\theta$  be the set of parameters of the model and let  $\theta_{ML}$  be their maximum likelihood estimate on the dataset  $\mathbf{X}$ . Then,

$$AIC = 2|\theta| - 2 \log p(\mathbf{X}|\theta_{ML}) = 2|\theta| - 2 \max_{\theta} l(\theta|\mathbf{X})$$

lower values correspond to models to be preferred.

**Bayesian Information Criterion (BIC)** A variant of the above, defined as

$$\begin{aligned} BIC &= |\theta| - \log |\mathbf{X}| 2 \log p(\mathbf{X} | \theta_{ML}) \\ &= |\theta| \log |\mathbf{X}| - 2 \max_{\theta} l(\theta | \mathbf{X}) \end{aligned}$$

## 2 Example: learning in the dirichlet-multinomial model

### Language modeling

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 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).

### Generative model

Given a language model, it is possible to sample from the distribution to generate random documents statistically equivalent to the documents in the collection used to derive the model.

### Language model

- Let  $\mathcal{T} = \{t_1, \dots, t_n\}$  be the set of terms occurring in a given collection  $\mathcal{C}$  of documents, after **stop word** (common, non informative terms) removal and **stemming** (reduction of words to their basic form).
- For each  $i = 1, \dots, n$  let  $m_i$  be the multiplicity (number of occurrences) of term  $t_i$  in  $\mathcal{C}$
- A language model can be derived as a categorical distribution associated to a vector  $\hat{\phi} = (\hat{\phi}_1, \dots, \hat{\phi}_n)^T$  of probabilities: that is,

$$0 \leq \hat{\phi}_i \leq 1 \quad i = 1, \dots, n \quad \sum_{i=1}^n \hat{\phi}_i = 1$$

where  $\hat{\phi}_j = p(t_j | \mathcal{C})$

### Learning a language model by ML

Applying maximum likelihood to derive term probabilities in the language model results into setting

$$\hat{\phi}_j = p(t_j | \mathcal{C}) = \frac{m_j}{\sum_{k=1}^n m_k} = \frac{m_j}{N}$$

where  $N = \sum_{i=1}^n m_i$  is the overall number of occurrences in  $\mathcal{C}$  after stopword removal.

## Smoothing

According to this estimate, a term  $t$  which never occurred in  $\mathcal{C}$  has zero probability to be observed (black swan paradox). Due to overfitting the model to the observed data, typical of ML estimation.

Solution: assign small, non zero, probability to events (terms) not observed up to now. This is called **smoothing**.

### Bayesian learning of a language model

We may apply the dirichlet-multinomial model:

- this implies defining a Dirichlet prior  $\text{Dir}(\phi|\alpha)$ , with  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$  that is,

$$p(\phi_1, \dots, \phi_n|\alpha) = \frac{1}{\Delta(\alpha_1, \dots, \alpha_n)} \prod_{i=1}^n \phi_i^{\alpha_i-1}$$

- the posterior distribution of  $\phi$  after  $\mathcal{C}$  has been observed is then  $\text{Dir}(\phi|\alpha')$ , where

$$\alpha' = (\alpha_1 + m_1, \alpha_2 + m_2, \dots, \alpha_n + m_n)$$

that is,

$$p(\phi_1, \dots, \phi_n|\alpha') = \frac{1}{\Delta(\alpha_1 + m_1, \dots, \alpha_n + m_n)} \prod_{i=1}^n \phi_i^{\alpha_i+m_i-1}$$

### Bayesian learning of a language model

The language model  $\hat{\phi}$  corresponds to the predictive posterior distribution

$$\begin{aligned} \hat{\phi}_j &= p(t_j|\mathcal{C}, \alpha) = \int p(t_j|\phi) p(\phi|\mathcal{C}, \alpha) d\phi \\ &= \int \phi_j \text{Dir}(\phi|\alpha') d\phi = E[\phi_j] \end{aligned}$$

where  $E[\phi_j]$  is taken w.r.t. the distribution  $\text{Dir}(\phi|\alpha')$ . Then,

$$\hat{\phi}_j = \frac{\alpha'_j}{\sum_{k=1}^n \alpha'_k} = \frac{\alpha_j + m_j}{\sum_{k=1}^n (\alpha_k + m_k)} = \frac{\alpha_j + m_j}{\alpha_0 + N}$$

The  $\alpha_j$  term makes it impossible to obtain zero probabilities (**Dirichlet smoothing**).

Non informative prior:  $\alpha_i = \alpha$  for all  $i$ , which results into

$$p(t_j|\mathcal{C}, \alpha) = \frac{m_j + \alpha}{\alpha V + N}$$

where  $V$  is the vocabulary size.

### Naive bayes classifiers

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

- 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  $\mathcal{C}_2$

## Naive bayes 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

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

### 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

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

By applying the product rule, it results

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

## 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  $i, j = 1 \dots, n_d$  and  $k = 1, 2$ ,

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

as, a consequence,

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

### Language models and NB classifiers

The probabilities  $p(\bar{t}_j|C_k)$  are available for all terms if language models have been derived for  $C_1$  and  $C_2$ , respectively from documents in  $\mathcal{C}_1$  and  $\mathcal{C}_2$ .

## Feature selection by mutual information

### Feature selection

The set of probabilities in a language model can be exploited to identify the most relevant terms for classification, that is terms whose presence or absence in a document best characterizes the class of the document.

### Mutual information

To measure relevance, we can apply the set of mutual informations  $\{I_1, \dots, I_n\}$

$$\begin{aligned} I_j &= \sum_{k=1,2} p(t_j, C_k) \log \frac{p(t_j, C_k)}{p(t_j)p(C_k)} \\ &= \sum_{k=1,2} p(C_k|t_j)p(t_j) \log \frac{p(C_k|t_j)}{p(C_k)} = p(t_j)KL(p(C_k|t_j)||p(C_k)) \end{aligned}$$

here,  $KL$  is a measure of the amount of information on class distributions provided by the presence of  $t_j$ . This amount is weighted by the probability of occurrence of  $t_j$ .

## Feature selection by mutual information

### Mutual information

Since  $p(t_j, C_k) = p(C_k|t_j)p(t_j) = p(t_j|C_k)p(C_k)$ ,  $I_j$  can be estimated as

$$\begin{aligned} I_j &= p(t_j|C_1)p(C_1) \log \frac{p(t_j|C_1)}{p(t_j)} + p(t_j|C_2)p(C_2) \log \frac{p(t_j|C_2)}{p(t_j)} \\ &= \phi_{j1}\pi_1 \log \frac{\phi_{j1}}{\phi_{j1}\pi_1 + \phi_{j2}\pi_2} + \phi_{j2}\pi_2 \log \frac{\phi_{j2}}{\phi_{j1}\pi_1 + \phi_{j2}\pi_2} \end{aligned}$$

where  $\phi_{jk}$  is the estimated probability of  $t_j$  in documents of class  $C_k$  and  $\pi_k$  is the estimated probability of a document of class  $C_k$  in the collection.

A selection of the most significant terms can be performed by selecting the set of terms with highest mutual information  $I_j$ .