Bayesian Classification

Introduction

A classification task may be described as the task of assigning a class label C to a given observation x.

The general approach lies in constructing *discriminant functions* $f(C|x)$ and let the winner decide to which class C sample x belongs.

Bayes Rule

A popular and simple implementation of a discriminant function is Bayes decision rule with a solid foundation in statistics. It states that the *posterior probabilities* $P(C|x)$ can serve as discriminant function and is even optimal under mild conditions.

Thus, our decision rule is finding the maximum of the posterior probabilities over all classes

$$C^* = max_C P(C|x)$$

with the posterior computed according to Bayes rule in following manner:

$$P(C|x) = \frac{p(x|C) \times P(C)}{p(x)}$$

in which:

- x is a feature Vector
- C is a class label
- $P(C|x)$ is the (posterior) probabilty for Class C given feature vector x
- $p(x|C)$ is the feature distribution of feature vector x for class C (Note: in discrete density models $p(x|C)$ is a probability and not a density)

Remark that the posterior probabilities in Bayes rule not only allow for classification but also give a measure for the accuracy of our decision.

Bayesian Classification in practice

which the density estimation translates into the estimation of the parameters of the distribution.

TRAINING PHASE

1. Collect the training data
2. Choose the model of the distributions
3. For each class: Estimate model parameters from the training data

RECOGNITION PHASE

1. get the class Priors $P(C)$ or neglect them if no prior knowledge is available
2. Compute class likelihoods $p(x|C)$ for all classes
3. Compute weighted likelihoods $p(x|C) \times P(C)$
4. Compute the total likelihood of the feature vector: $p(x) = \sum_C p(x|C) \times P(C)$
5. Compute the posteriors: $P(C|x)$
6. Take the maximum over the posteriors

Note: The total likelihood of the sample is just a normalization factor guaranteeing that the posteriors sum up to 1.0. This step is not necessary for classification as such, but inspecting the posteriors is often a good sanity check on your results.

Optimality and Limitations for Bayesian Classification

Bayes decision rule will be optimal (i.e. no better decision rule can be constructed!!) under following conditions:

- you need enough (correctly) labeled training data such that you can estimate class distributions $p(x|C)$ using maximum likelihood estimation
- you need to know prior class probabilities $P(C)$
- you should use Bayes rule (about conditional probabilities) to compute the posterior probabilities

These conditions look mild at first sight. However, experience has taught us that the Bayesian approach has significant limitations for complex problems. The central issue is the estimation of the class densities $p(x|C)$. For real life data such densities will not conform to a simple shape like Gaussian or binomial for which maximum likelihood estimation has a straightforward solution. On the contrary more often than not we need to make use of so called universal approximators such as Gaussian Mixture Models (GMMs). These mixture distributions are approximate at best and a maximum likelihood estimation

concerned. But even more: how much better do our estimates get, and consequently the decision rule, with increasing amounts of data. The truth is, improvement is inherently slow with increasing amounts of data.

Looking at the formulas in detail gives us insight into the inherent underlying problem. For a sample that scores high on one class and not so high on the others, there will be no no problem; small errors in the estimated probabilities will not influence the classification outcome. But for outliers, i.e. samples that aren't modeled well by any of the classes, there is a fundamental problem. All class densities $p(x|C)$ are very small in this case. The challenge is thus to discriminate between *very very small* on one class vs. *extremely small* on another class. Estimating ϵ-small numbers will never be easy and comparing many of them is even more challenging. Another way of looking at this: the posterior computation is based on a 0/0 division !!! This contrasts strongly with the fact at least one of the posteriors $P(C|x)$ must be finite. ... unless the outlier doesn't really belong to any class and you forgot to include and/or model such dummy outsider class.

Generative vs. Discriminative models

In the case of the Bayesian approach we use a **Generative** model: i.e. we estimate the model distributions (i.e. a model that can generate artificial data). The **discriminant** functions are computed indirectly from the **generative** model.

Modern **Deep Neural Nets** are **Discriminative** models as they estimate the discriminant functions directly by minimizing the classification error on a given train set (or optimize another criterion that is directly related to classification). Often it is the case the discriminant functions are normalized and that they can be interpreted as posteriors.

Generative and discriminative models each have a number of advantages and disadvantages:

- In the generative model, the feature distributions can be estimated for each class independently. This is convenient because you can add / split classes at will. It is also less demanding because a global problem (classification) is split into N subproblems (density estimations) that can be solved independent of each other.
- In a discriminative model, the optimization is global. The discriminant functions are learned jointly using a single overall optimization function. This optimization will inherently be an order of magnitude more complex than in the case of a generative model.
- Given the higher complexity, discriminative models will at the same time need and also benefit from a large training corpus, again pushing the computational requirements up.

In summary **Generative Models** can be trained quickly with small amounts of data and may be the preferred solution in the case of limited resources, inherently sparse data problems, prototyping, ... **Discriminative Models** are superior when large representative corpora are available for your problem. They may also be the methodology of choice if adapatation or fine tuning is an option in which some large background model can serve as reference.

Gaussian Mixtures as universal approximators

$$p(x|C_k) = \sum_{j=1}^{M} w_{ij} \mathcal{N}(x; \mu_{kj}, \Sigma_{kj})$$

in which $\mathcal{N}_{kj}$ is the j-th mixture of Class C_k parameterized by $w_{kj}, \mu_{kj}, \Sigma_{kj}$, respectively the weight, mean and convariance matrix. Without any constraint on the parameters, these functions are also known as Radial Basis Functions. In the probabilistic literature GMMs are used as probability density functions. This merely requires that the weights sum up to 1.

The parameters of a single Gaussian are easily estimated from example data using the maximum likelihood principle. For the 1D case, yielding:

$$\hat{\mu} = \frac{1}{N}\sum_i x_i$$

$$\hat{\sigma^2} = \frac{1}{N-1}\sum_i (x_i - \mu)^2$$

Estimating parameters of a Gaussian Mixture model from data is more involved and only approximate. The *EM* (Estimation-Maximization)* algorithm finds a local optimum in an iterative way.

Standard Normal Distribution

The z-transformation

$$z = \frac{x - \mu}{\sigma}$$

measures the number of standard deviations that a random variable deviates from the mean. The z-transformation may be used to map any normal (Gaussian) distribution to the *standard normal* distribution with probability density function (pdf):

$$p(z) = \frac{1}{\sqrt{2\pi}} exp\left(-\frac{1}{2}z^2\right)$$

The figures below show both pdf (above) and the cumulative density function (cdf) , i.e. the probability mass under the curve to the left of a given point.