An EM Algorithm for Fitting Maximum Likelihood Linear Models

with Gaussian Densities for Classification

Samuel Otto

**Introduction**

**Derivation of Algorithm**

Suppose we have some nonlinear function of the following form

We have in mind a discrete or continuous time dynamical system with control input . Suppose we have collected training data . We wish to approximate the function by a collection of affine linear models spread throughout the domain. In addition to the models themselves, we require a way to assign a new point to the nearby model which will best approximate its dynamics under . Let there be linear models of the form

where is Gaussian noise having covariance . Let be the random variable whose value is the model describing the dynamics at the training point . Assume the probability density that the dynamics of the point are modeled by the th linear model is also Gaussian in the spatial coordinates

Assume that the prior probabilities follow some Markov transition process

We wish to find the parameters of which maximize the likelihood of the training data

We know from our assumptions that

Therefore, the problem is in a form to which the EM algorithm is directly applicable. We quickly review the EM algorithm as it applies to the problem of maximizing the above likelihood function. Find the log likelihood

Introduce a collection of yet to be determined distributions over the possible models for each training point .

The inside term is now in the form of an expectation

By Jensen’s inequality and the concavity of the logarithm, we observe that

Substitute the known forms of the distributions

Let be a previous set of parameters for the collection of linear models. The distribution is set to be the normalized probability

This choice is made so that the above inequality becomes equality if and only if . One can see that this must be the case since the term inside the expectation becomes independent of when and therefore . The EM algorithm proceeds by repeating the following two steps until convergence

E-step: Set the nominal value and find

M-step: Maximize the lower bound on the likelihood using the above expectation

Subject to the constraints

Let us proceed with the maximization by splitting up the objective

Let

Maximizing the second term is analogous to the well-known mixture of Gaussians model so we will carry out this maximization first

Solve for the centroids of the spatial Gaussians

Solve for the spatial covariance matrices.

Let

Then

Now we will proceed to maximize the first term

The result from above immediately gives

We now must find the matrices and in addition to . This is the classic weighted linear regression problem with weights given by . Let

Then

and we will try to find , and all at the same time by differentiating with respect to

Let

Then

It should be noted that our observables are not limited to be linear functions. By choosing to include quadratic and higher terms, we can build a mixture of nonlinear models instead of linear ones. The second order terms may be used to estimate the error due to bias in the model. It is also possible to formulate the nonlinear version of the algorithm using regularized kernel regression. Finally, we maximize the third term subject to the constraint by forming the Lagrangian

Summing over and using the constraint we have

Therefore,

We now solve for the prior distribution

Sum over and use the constraint

Therefore

**Reconstructing Estimates**

Once the linear models are fit to the training data, the dynamics of a new point can be estimated using the collection of linear models and their densities in a number of ways. We start by approximating the prior probabilities of the models

Now the posterior probability is found using the knowledge of the current state

Therefore, we find the probability distribution over the models at point

Here we will present three possibilities for estimating the next state using the above distribution over the models at the current point

1. The simplest method is to choose the maximum likelihood linear model at
2. The first method allows for approximations which are discontinuous. If we desire continuity then a natural method is to combine some or all linear models at a given point according to their likelihood. Hence,
3. Finally, we may choose which model to apply at stochastically according to the above weights

This is easily accomplished by letting be a random variable sampled from a uniform distribution over . Then let

Choosing the linear models stochastically gives us a generative probabilistic model of the system’s dynamics which may be useful in some cases. In such situations, the statistical properties of the dynamics are of greater interest than accurate prediction of the next state.

**Implementation Overview**

Form data matrices

Initialize variables

Choose randomly in training data . Initialize the Markov process variables using uniform distributions

Initialize the joint probability for calculating the expectation

Loop until convergence {

Expectation step:

Maximization step

Loop over models {

Let

Let

Let

} End loop over models

Loop over {

Update priors

Update joint probability for calculating the expectation

Loop over models {

} End loop over models

} End loop over points

} End EM convergence loop