+++++++++++++++++

In practical, many information providing useful cues for prediction is not directly observable from data. For motif (repeated patterns in DNA sequences) finding problem, as an example, our task if to find motifs from a set of DNA sequences where the location of these motifs are unknown. Thus the information of position can be treated as hidden variable and is important to be considered in the model though it is not directly observable. The same problem raises in computer vision when we wish to learn a model from images lacking of additional annotated information. Issues like this have been long studied by many researchers and latent SVMs, which can explicitly model hidden variables with joint feature vectors, outperforms many other methods.

The latent SVM was developed by Felzenszwalb et al.\cite and Yu and Joachims\cite independently in different ways. The main idea is introducing a latent variable to extend the feature vector, which results in a generalized loss function, e.g. Hinge Loss, with an upper bound. Then the optimization was done by using Concave-Convex Procedure (CCCP) algorithm.

intro-latentposition

-------------------------

Latent variable models (lvm) provide an elegant formulation for several applications of practical importance. For example, in computer vision, we may wish to learn a model of an object category such as ‘car’ from images where the location of the car is unknown, and is therefore treated as a latent (or hidden) vari- able. In computational medicine, we may wish to di- agnose a patient based on the observed symptoms as well as other unknown factors—represented using hid- den variables—such as the family’s medical history.

An lvm consists of three types of variables: (i) the observed variables, or input, whose values are known during both training and testing; (ii) the unobserved variables, or output, whose values are known only dur- ing training; and (iii) the hidden variables, whose val- ues are unknown during both training and testing. An lvm models the distribution of the output and hidden variables conditioned on, or jointly with, the input. Modeling the conditional distribution results in discriminative lvms, while modeling the joint distribu- tion results in generative lvms. Given an input, the output is typically predicted by either (i) computing the most probable assignment of the output and the hidden variables according to the aforementioned distribution [5, 25]; or (ii) computing the most probable assignment of the output by marginalizing out the hid- den variables [4]. Both these prediction criteria ignore an important factor: how certain are we about the values of the hidden variables for the predicted out- put? Since the underlying assumption of lvm is that the hidden variables provide useful cues for predicting the output, we argue that minimizing the confusion in their values will help improve the accuracy of the model. Furthermore, in many cases there is value in obtaining an estimate of the hidden variables them- selves. For example, using an lvm for a ‘car’ we would like not only to classify an image as containing a car or not, but also predict the location of the car if present.

chunNam-experi-motifs-19

-------------------

Latent structured SVMs outperform SVMs on problems where they can explicitly model problem-dependent hidden variables. A popular example task is the binary classification of pro- tein DNA sequences [18, 20, 19]. The hidden variable to be modeled is the unknown location of particular subsequences, or *motifs*, that are indicators of positive sequences.