Generative vs. Discriminative

Jeremy Irvin and
Daniel Spokoyny

Created from Andrew Ng's Stanford CS229 Notes and Coursera Stanford NLP

Introduction

- So far we've mostly talked about learning algorithms which model $p(y|x;\theta)$, the conditional distribution of *y* given *x*.

- For instance, logistic regression modeled $p(y|x;\theta)$ as $h_\theta(x) = g(\theta^T x)$, where *g* is the sigmoid function.

The Old Approach

- Consider a classification problem where we want to learn to distinguish between whether a house will fall of DP (y=*1*) or not (y=*0*), based on some features of the house (and its parties).

- Logistic regression tries to learn a straight line (a decision boundary) to separate the houses.

- Then to classify which house will fall off the cliff, it checks which side of the decision boundary the house lies, and predicts accordingly.

A New Approach

- Suppose instead we first look at the houses which have fallen off and build a model of what these houses look like.

- Then we look at the houses which are stable and build a model of what these houses look like.

- Finally, to classify a house, we can match the house against both models and see which best models it.

Discriminative vs. Generative

- Algorithms which try to learn $p(y|x)$ directly (such as logistic regression) or algorithms which try to learn mappings from the space of inputs to the labels {0,1} (such as the perceptron algorithm which we will discuss later) are called <u>discriminative</u> algorithms.

- Algorithms which try to model $p(x|y)$ (and $p(y)$) are called <u>generative</u> algorithms.

- In the our example, $p(x|y=0)$ models the distribution of stable house *features*, and $p(x|y=1)$ models the distribution of unstable house features.

Bayes Theorem!!!

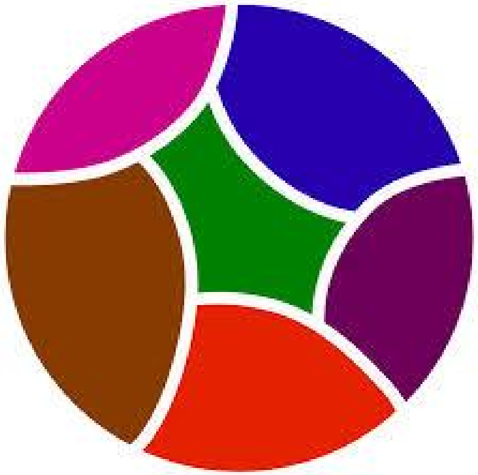
- If A and B are any two events, then

$$P(A|B) = \frac{P(A)\,P(B|A)}{P(B)}$$

- If {*A_j*} is a partition of the sample space, then

$$P(B) = \sum_j P(B \mid A_j) P(A_j),$$

$$\Rightarrow P(A_i \mid B) = \frac{P(B \mid A_i)\,P(A_i)}{\sum_j P(B \mid A_j)\,P(A_j)}.$$



Discriminative vs. Generative

- After modeling $p(y)$ (called the class priors) and $p(x|y)$, we can then use Bayes rule to derive the posterior distribution of *y* given *x*:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} = \frac{p(x|y)p(y)}{p(x|y=1)p(y=1) + p(x|y=0)p(y=0)}$$

- Hence everything can be computed using the quantities we have modeled. But we don't really need to compute the denominator since:

$$\begin{aligned} \arg\max_y p(y|x) &= \arg\max_y \frac{p(x|y)p(y)}{p(x)} \\ &= \arg\max_y p(x|y)p(y). \end{aligned}$$

Summary

- Generative classifiers learn a model of the joint probability $p(x, y) = p(x|y)p(y)$, of the inputs and label, and make their predictions by using Bayes rule to calculate $p(y|x)$.

- Discriminative classifiers model the posterior $p(y|x)$ directly, or learn a direct map from inputs to the labels.

- So a parametric family of probabilistic models $p(x, y)$ can be fit either to optimize the joint likelihood of the inputs and labels or the conditional likelihood.

Important Note

- A pair of classifiers, one discriminative and one generative which use the same parametric family of models, is called a Generative-Discriminative pair.

- For example, if $p(x|y)$ is Gaussian and $p(y)$ multinomial, then the Gen-Dis pair is Normal Discriminant Analysis and logistic regression.

- In the discrete case, the naive Bayes classifier and logistic regression form a Generative-Discriminative pair.