

## CS134 – Handout on Maximum Entropy Models

This handout walks through a very simple example of computing the posterior distribution (inference) using a Maximum Entropy model and using this posterior to compute the expected values of each feature in the model. These expectations allow us to compute the components of the gradient required for learning.

Assume the following four feature functions in our model:

$$f_1(x, y) = 1 \text{ if } y = \text{Yes} \ \&\& \ x = \text{Sunny}$$

$$f_2(x, y) = 1 \text{ if } y = \text{No} \ \&\& \ x = \text{Sunny}$$

$$f_3(x, y) = 1 \text{ if } y = \text{Yes} \ \&\& \ x = \text{Rain}$$

$$f_4(x, y) = 1 \text{ if } y = \text{No} \ \&\& \ x = \text{Rain}$$

Let's assume the lambda values (i.e. the parameters) for each of these four features are as follows:

$$\lambda_1 = 1.1$$

$$\lambda_2 = -1.1$$

$$\lambda_3 = -1.5$$

$$\lambda_4 = 1.5$$

Let's assume we're given the following data set:

$$D = \{(S, Y), (S, Y), (R, N), (R, N), (S, N), (R, Y)\}$$

Given the above parameters, we have the following posterior distributions (inference):

$$P(y|x) = \frac{\exp(\sum_k \lambda_k f_k(x, y))}{\sum_z \exp(\sum_k \lambda_k f_k(x, z))}$$

Which works out to:

$$P(y = Y|x = S) = \frac{\exp(1.1)}{\exp(1.1) + \exp(-1.1)} = \frac{3.0042}{3.0042 + 0.3329} = 0.9002$$

$$P(y = Y|x = R) = \frac{\exp(-1.5)}{\exp(1.5) + \exp(-1.5)} = \frac{0.2231}{4.4817 + 0.2231} = 0.0474$$

Let's first compute the gradient components where each is defined as:

$$\sum_{i=1}^{|D|} f_k(x, y) - \sum_{i=1}^{|D|} \sum_z p(z|x) f_k(x, z)$$

For the first feature/parameter, the empirical expectation is 2.0 (from the left summand). Let's work out the model expectation for each data exemplar separately:

When  $i=1$ , we have we need to consider  $p(z = Y|x = S)f_1(x, z) + p(z = N|x = S)f_1(x, z)$ . As this feature only returns 1.0 when  $z=Y$  and  $x=S$ , we have just  $p(z = Y|x = S)f_1(x, z)$  which works out to just  $p(z = Y|x = S)$  which is 0.9002. We'll end up with this same value when  $i=1, i=2, i=5$  (as only those examples have  $x=S$ ) and 0.0 for  $i=3, 4$  and 6. This leaves us with an expected value for  $f_1$  of:

$$0.9002 + 0.9002 + 0.9002 = 2.7006$$

It's important to note that when we compute the empirical expectations, we must pay attention to the values of the dependent variable that we're trying to predict (i.e. 'y'). When we compute the expectations, we're summing over all values of 'y' for each data instance.

The gradient component for the first feature/parameter is thus  $2.0 - 2.7006 = -0.7006$

Let's look at the second feature,  $f_2$ , its empirical expectation is: 1.0 while the model expectation is:

$$.0998 + .0998 + .0998 = .2994$$

Thus, the gradient component is  $1.0 - 0.2994 = .7006$

The third feature,  $f_3$ , has an empirical expectation of 1.0 and a model expectation of:

$$0.0474 + 0.0474 + 0.0474 = 0.1422, \text{ so the gradient component is } 0.8578$$

For  $f_4$ , we have an empirical expectation of 2.0 and the model expectation is:

$$0.9526 + 0.9526 + 0.9526 = 2.8578, \text{ the gradient component is thus: } 2.0 - 2.8578 = -0.8578$$

The entire gradient is thus:

$$[-0.7006, 0.7006, 0.8578, -0.8578]$$

Think a bit about what the maximum likelihood solution should look like based on our data. What should we expect the posterior distribution  $P(y|x = S)$  to look like (given that there two cases where  $y=Y$  and one where  $y=N$ )?

## Learning Process Outline:

As discussed in lecture, the learning process with Maximum Entropy models is an iterative process. The following steps are repeated until the training process converges. That is, until the model parameters converge to the maximum likelihood solution.

- 1) Initialize parameters (usually to 0.0)
- 2) With the current parameters,  $\Lambda^{(t)}$  compute the log-likelihood,  $LL_{\Lambda}$ , and the gradient of the log-likelihood,  $\nabla LL_{\Lambda}$

- 3) Derive the next set of parameters using an appropriate optimization algorithm:

$$\Lambda^{(t+1)} \leftarrow \text{Update}(\Lambda^{(t)}, LL_{\Lambda}, \nabla LL_{\Lambda})$$

- 4) Go to (2) unless converged