CSC 446 Notes: Lecture 16

prepared by Phyo Thiha

April 9, 2012

1 Maximum Entropy (Logistic Regression or Log-Linear)

$$P(y|x) = \frac{1}{Z_x} e^{\sum_i \lambda_i f_i(x,y)}$$

$$Z_x = \sum_y e^{\sum_i \lambda_i f_i(x,y)}$$

e.g., In NLP, we may use

$$f_{100}(x,y) = \begin{cases} 1 & \text{if x=word that ends with 'tion' and y=Noun}\,, \\ 0 & \text{if otherwise}\,. \end{cases}$$

The above is more general binary classification where we are only deciding whether an example belongs to a class or NOT. Here, we can have features contributing to multiple classes according to their weights. For binary classification, the decision boundary is linear, as with perceptron or SVM. A major difference from SVMs is that, during training, every example contributes to the objective function, whereas in SVMs only the examples close to the decision boundary matter.

If we plot this function, we get a sigmoid-like graph. We can draw analogy between maximum entropy and neural network, and consider features as the input nodes in the neural network.

If we take the log of equation (1), we get a linear equation

$$\log P = \sum_i \lambda_i f_i + c$$

What should λ be?

$$\begin{aligned} \max_i \ \log \left(\prod_{n=1}^{N} \ P(y_n|x_n) \right)^{\frac{1}{N}} &= \max \sum_n \frac{1}{N} \ \log \left(\frac{1}{Z_x} \ e^{\sum_i \lambda_i f_i} \right) \\ &= \frac{1}{N} \sum_n \ \left(\sum_i \lambda_i f_i - \log Z_x \right) \end{aligned}$$

In the above equation, maximizing λ_i is easy, but maximizing Z_x is not. To maximize for λ_i, we turn it into a concave form and find the point where the derivative w.r.t. λ is zero (hill climbing)

$$L = \frac{1}{N}\sum_n \sum_i \lambda_i f_i - \log \sum_y e^{\sum_i \lambda_i f_i(x_n, y)} \tag{1}$$

$$\frac{\partial L}{\partial \lambda_j} = \frac{1}{N}\sum_n f_j - \frac{1}{Z_x}\frac{\partial}{\partial \lambda_j}(\sum_y e^{\sum_i \lambda_i f_i}) \tag{2}$$

$$= \frac{1}{N}\sum_n \left[f_j - \frac{1}{Z_x}\sum_y f_j e^{\sum_i \lambda_i f_i(y, x_n)} \right] \tag{3}$$

$$= \frac{1}{N}\sum_n f_j - \sum_y f_j P\left(y|x_n\right) \tag{4}$$

$$= \frac{1}{N}\sum_n f_j(x_n, y_n) - \sum_y f_j P\left(y|x_n\right) \tag{5}$$

We can morph (6) into expectation form by defining joint probability as follows:

$$P(y, x) = P\left(y|x\right)\overline{P}(x) \tag{6}$$

$$\overline{P}(x) = \frac{1}{N}\sum_n I(x_n = x) \tag{7}$$

$$\overline{P}(y|x) = \frac{c(x_n = x, y_n = y)}{c(x_n = x)} \tag{8}$$

Rewriting equation(6) in expectation form, we get:

$$\frac{\partial L}{\partial \lambda_j} = E_{\overline{P}}[f_j] - E_P[f_j] \tag{9}$$

where the first term (before the minus) is a constant, and the complexity of calculating the second term depends on the number of classes in the problem. Now we have:

$$\lambda \leftarrow \lambda + \eta \frac{\partial L}{\partial \lambda} \tag{10}$$

We will justify why we chose log linear form instead of something else. Assume we want to find the maximum entropy subject to constraints on the feature expectations:

$$\max_{P(y|x)} H(y|x) \tag{11}$$

$$\min_{P(y|x)} -H(y|x) \tag{12}$$

$$\text{s.t. } E_{\overline{P}}[f_i] - E_P[f_i] \ \forall i \tag{13}$$

$$\sum_y P(y|x) = 1 \ \forall x \tag{14}$$

To put this into words, we want to build a model such that for each feature, our model should match the training data. We have

$$H(y|x) = \sum_{x,y} \overline{P}(x)P(y|x) \log \frac{1}{P(y|x)} \tag{15}$$

Find the maximum entropy of the above equation as follows

$$\begin{aligned} L(P, \lambda, \mu) =& f_0 + \sum_j \lambda_j f_i & (16)\\ =& \sum_{x,y} \overline{P}(x)P(y|x) \log P(y|x) & (17)\\ &+ \sum_i \lambda_i \left(\sum_{x,y} \overline{P}(x)\overline{P}(y|x)f_i - \overline{P}(x)P(y|x)f_i \right) & (18)\\ &+ \sum_x \mu_x \left(\sum_y P(y|x) - 1 \right) & (19) \end{aligned}$$

$$\frac{\partial L}{\partial P(y|x)} = \overline{P}(x) \left(\log P(y|x) + 1 \right) + \sum_i \lambda_i \overline{P}(x) f_i + \mu_x = 0 \tag{20}$$

$$\begin{aligned} \log P(y|x) &= -1 + \sum_i \lambda_i f_i - \frac{\mu_x}{\overline{P}(x)} & (21)\\ P(y|x) &= e^{-1 - \frac{\mu_x}{\overline{P}(x)}} e^{\sum_i \lambda_i f_i} & (22)\\ &= \frac{1}{Z_x} e^{\sum_i \lambda_i f_i} & (23) \end{aligned}$$

The above result shows that maximum entropy has log-linear form. If we solve the dual of the problem

$$\max_{\lambda,\mu} g(\lambda,\mu) = E_P\left[\sum_i \lambda_i f_i - logZ_x\right] + E_{\overline{P}}\left[\sum_i \lambda_i f_i\right] - E_P\left[\sum_i \lambda_i f_i\right] \tag{24}$$

$$= E_P\left[-\log Z_x\right] + E_{\overline{P}}\left[\sum_i \lambda_i f_i\right] \tag{25}$$

$$= E_{\overline{P}}\left[-\log Z_x\right] + E_{\overline{P}}\left[\sum_i \lambda_i f_i\right] \tag{26}$$

$$= E_{\overline{P}}\left[\sum_i \lambda_i f_i - \log Z_x\right] \tag{27}$$

$$= E_{\overline{P}} \log P(y|x) \tag{28}$$

$$= \frac{1}{N}\sum_n \log P(y_n|x_n) \tag{29}$$

The result from above is the problem we started with. This justifies why we use $\frac{1}{Z_x} e^{\sum_i \lambda_i f_i}$ form for maximum entropy.