Notes on Structured Prediction

Chunpai Wang
chunpaiwang@gmail.com

June 18, 2019

Contents

1 Conditional Random Fields and Structured SVM

Given a set of i.i.d sampled training data $\mathcal{D} = \{(x^m, y^m)\}_{m=1,\cdots,M}$, where $(x^m, y^m) \sim d(x, y)$, $x^m \in [0,1]^N$, and $y \in \{0,1\}^N$. A set of feature functions are provided such as $\phi(x,y) = (\phi_1(x,y), \cdots, \phi_K(x,y))$. Our task is to find a good parameterized distribution

$$p(y|x,w) = \frac{1}{Z(x,w)} \exp(\langle w, \phi(x,y)\rangle) \tag{1}$$

that well approximate the true distribution $d(y|x)$, where we can adjust the parameter $w \in \mathbb{R}^K$ based on the training data $\mathcal{D}$. $Z(x,w)$ is known as the partition function

$$Z(x,w) = \sum_{y \in Y} \exp(\langle w, \phi(x,y)\rangle) \tag{2}$$

since it is intractable to do the summation over all possible $y \in Y$, we have to use the structural information.

With the maximum likelihood parameter estimation, we can derive the optimal w as

$$\begin{aligned}
w^* &= \arg\max_w p(y^1, \cdots, y^M \mid x^1, \cdots, x^M; w) && (3)\\
&= \arg\max_w \prod_{m=1}^{M} p(y^m|x^m; w) && (4)\\
&= \arg\min_w - \sum_{m=1}^{M} \log p(y^m|x^m; w) && (5)\\
&= \arg\min_w - \sum_{m=1}^{M} \log \left[\frac{1}{Z(x^m, w)} \exp(\langle w, \phi(x^m, y^m)\rangle)\right] && (6)\\
&= \arg\min_w - \sum_{m=1}^{M} [\langle w, \phi(x^m, y^m)\rangle - \log Z(x^m, w)] && (7)\\
&= \arg\min_w - \sum_{m=1}^{M} \left[\langle w, \phi(x^m, y^m)\rangle - \log \sum_{y \in Y} \exp(\langle w, \phi(x,y)\rangle)\right] && (8)\\
& && (9)
\end{aligned}$$

Alternatively, with the maximum a posterior estimation, we can derive the optimal w as

$$\begin{aligned}
w^* &= \underset{w \in \mathbb{R}^K}{\text{argmax}}\; p(w|\mathcal{D}) && (10)\\
&= \underset{w \in \mathbb{R}^D}{\text{argmin}} [-\log p(w|\mathcal{D})] && (11)\\
&= \underset{w \in \mathbb{R}^D}{\text{argmin}} \left[-\log \prod_{m=1}^{M} \frac{p(y^m|x^m; w) \cdot p(w)}{p(y^m|x^m)}\right] && (12)\\
&= \underset{w \in \mathbb{R}^D}{\text{argmin}} \left[-\log p(w) - \sum_{m=1}^{M} \log p\left(y^n|x^m, w\right) + \log p\left(y^m|x^n\right)\right] && (13)\\
&= \underset{w \in \mathbb{R}^D}{\text{argmin}} \left[-\log p(w) - \sum_{m=1}^{M} \log p\left(y^m|x^m, w\right)\right] && (14)
\end{aligned}$$

If we set the prior distribution $p(w)$ as constant, then the optimal w is same as the MLE. If we set the prior distribution as Gaussian:

$$p(w) = const \cdot \exp(-\frac{1}{2\sigma^2}\|w\|_2^2) \tag{15}$$

then we have the optimal w as

$$w^* = \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 - \sum_{m=1}^{M} \log p(y^m|x^m;w) \tag{16}$$

$$= \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 - \sum_{m=1}^{M}\left[\langle w, \phi(x^m, y^m)\rangle - \log \sum_{y\in Y} \exp(\langle w, \phi(x^m, y)\rangle)\right] \tag{17}$$

$$= \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 + \sum_{m=1}^{M}\left[\log \sum_{y\in Y} \exp(\langle w, \phi(x^m, y)\rangle) - \log \exp(\langle w, \phi(x^m, y^m)\rangle)\right] \tag{18}$$

$$= \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 + \sum_{m=1}^{M}\left[\log \frac{\sum_{y\in Y} \exp(\langle w, \phi(x^m, y)\rangle)}{\exp(\langle w, \phi(x^m, y^m)\rangle)}\right] \tag{19}$$

$$= \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 + \sum_{m=1}^{M}\left[\log \sum_{y\in Y} \frac{\exp(\langle w, \phi(x^m, y)\rangle)}{\exp(\langle w, \phi(x^m, y^m)\rangle)}\right] \tag{20}$$

$$= \arg\min_w \frac{1}{2\sigma^2}\|w\|_2^2 + \sum_{m=1}^{M}\left[\log \sum_{y\in Y} \exp(\langle w, \phi(x^m, y)\rangle - \langle w, \phi(x^m, y^m)\rangle)\right] \tag{21}$$

$$\tag{22}$$

On the other hand, the objective of the structured SVM with the re-scaled soft margin can be formulated as [1]

$$\min_w \frac{1}{2}\|w\|_2^2 + \frac{C}{M}\sum_{m=1}^{M}\left[\max_{y\in Y} \triangle(y^m, y) + \langle w, \phi(x^m, y)\rangle - \langle w, \phi(x^m, y^m)\rangle)\right] \tag{23}$$

where $\max_{y\in Y} \triangle(y^m, y) + \langle w, \phi(x^m, y)\rangle - \langle w, \phi(x^m, y^m)\rangle)$ is known as the hinge loss, which is a convex but non-differentiable surrogate loss function. We can see that the CRF and StructSVM have more in common than usually assumed. The $\log \sum_{y\in Y} \exp$ can be interpreted as a soft-max. The essential difference is that, StructSVM is cost-augmented i.e. $\triangle(y^m, y)$, but CRF is not.

2 Training Criteria

We review several criteria for training the weights w, including the conditional log-likelihood, max-margin, and risk augmented [4].

2.1 Conditional Log Likelihood

The trivial objective is to minimize the negative conditional log-likelihood without regularization:

$$\text{NCLL}: \quad \min_w - \sum_{m=1}^{M}\left\{\langle w, \phi(x^m, y^m)\rangle - \log \sum_{y\in Y} \exp(\langle w, \phi(x^m, y)\rangle)\right\} \tag{24}$$

2.2 Max-Margin

$$\begin{aligned} \text{MM}: \quad & \min_w \frac{C}{M}\sum_{m=1}^{M}\left[\max_{y\in Y} \triangle(y^m, y) + \langle w, \phi(x^m, y)\rangle - \langle w, \phi(x^m, y^m)\rangle)\right] \\ & \min_w - \sum_{m=1}^{M}\left\{\langle w, \phi(x^m, y^m)\rangle) - \max_{y\in Y} \left(\langle w, \phi(x^m, y) + \triangle(y^m, y)\rangle\right)\right\} \end{aligned} \tag{25}$$

2.3 Risk-Augmented

The risk or reward augmented objective is motivated by the objective of reinforcement learning [5].

$$\text{Risk}: \quad \min_w \sum_{m=1}^{M}\left\{\sum_{y\in\mathcal{Y}} p(y|x^m;w)\cdot \triangle(y, y^m)\right\} \tag{26}$$

where in risk minimizatoin $\triangle(y, y^m)$ denotes the cost or difference between y and y^m, which typically is assumed to be non-negative; We want to adjust w such that the probability of $p(y|x^m; w)$ is small if the cost is large. Unlike previous two objectives, the risk is typically non-convex. In addition, the computation of gradient of risk w.r.t w is challenging.

2.4 Soft Max-Margin

We see that the conditional log-likelihood is not cost augmented, and the $\log \sum \exp$ can be interpreted as soft max operation. It is intuitive to make it aligned with the max-margin objective by augmenting a cost.

$$\text{SoftMM} : \quad \min_w - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log \sum_{y \in Y} \exp(\langle w, \phi(x^m, y) \rangle + \triangle(y^m, y)) \right\} \tag{27}$$

All the objectives above are called the surrogate loss functions, which are used as the proxy of the $0-1$ empirical loss. Here, we will discuss the difference between the Soft Max-Margin and other objectives. We will show that:

- Soft Max-Margin is the convex differential upper bound of the max-margin loss, since the soft-max is a smooth upper bound of max operation.
- Soft Max-Margin is the convex upper bound of the NCLL and Risk objectives. We first denote

$$Z_m = \sum_{y \in Y} \exp(w^\top \phi(x^m, , y)) \tag{28}$$

$$\begin{aligned}
\text{Soft MaxMargin} &= - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log \sum_{y \in Y} \exp(\langle w, \phi(x^m, y) \rangle + \triangle(y^m, y)) \right\} && (29) \\
&= - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log \left[Z_m \cdot \sum_{y \in Y} \frac{\exp(\langle w, \phi(x^m, y) \rangle + \triangle(y^m, y))}{Z_m} \right] \right\} && (30) \\
&= - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log \left[Z_m \cdot \sum_{y \in Y} \frac{\exp(\langle w, \phi(x^m, y) \rangle)}{Z_m} \cdot \exp(\triangle(y^m, y)) \right] \right\} && (31) \\
&= - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log \left[Z_m \cdot \sum_{y \in Y} p(y|x^m; w) \cdot \exp(\triangle(y^m, y)) \right] \right\} && (32) \\
&= - \sum_{m=1}^{M} \left\{ \langle w, \phi(x^m, y^m) \rangle - \log Z_m - \log \left[\cdot \sum_{y \in Y} p(y|x^m; w) \cdot \exp(\triangle(y^m, y)) \right] \right\} && (33) \\
&= \underbrace{- \sum_{m=1}^{M} \{ \langle w, \phi(x^m, y^m) \rangle - \log Z_m \}}_{NCLL} + \sum_{m=1}^{M} \left\{ \log \mathbb{E}_{p(y|x^m;w)} \left[\exp(\triangle(y^m, y)) \right] \right\} && (34) \\
& && (35)
\end{aligned}$$

Since log is concave, we can use the Jensen's inequality to obtain, for all m:

$$\log \mathbb{E}_{p(y|x^m;w)} \left[\exp(\triangle(y^m, y)) \right] \geq \mathbb{E}_{p(y|x^m;w)} \left[\log \exp(\triangle(y^m, y)) \right] = \mathbb{E}_{p(y|x^m;w)} \left[\triangle(y^m, y) \right] = \text{Risk} \tag{36}$$

Since NCLL≥ 0 and $\log \mathbb{E}_{p(y|x^m;w)} \left[\exp(\triangle(y^m, y)) \right] \geq 0$ (assumed $\triangle(y^m, y) \geq 0$), Soft Max-Margin is convex upper bound of the objectives of NCLL and Risk. The computation of the gradient of Soft MaxMargin w.r.t w is more easier than using the Risk, we may consider to use this as upper bound of risk to find a good solution of risk minimization.

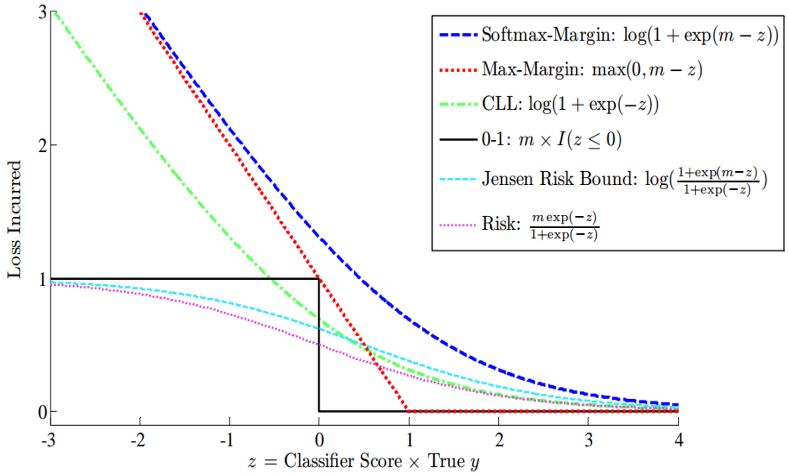

Figure 1: Surrogate Loss Functions for Binary Classification. m is the multiplier for the cost associated with making wrong classification decision; here $m = 1$. x-axis means the $f(x) \cdot y^*$

.

3 Reward Augmented Maximum Likelihood

As stated in previous section, maximizing the negative conditional log-likelihood on training data D will make all negative outputs are equally wrong, and none is preferred over the others.

$$\mathcal{L}_{\mathrm{ML}}(\boldsymbol{w}; \mathcal{D}) = \sum_{(\mathbf{x^m}, \mathbf{y}^m) \in \mathcal{D}} -\log p\left(\mathbf{y}^m | \mathbf{x}^m; w\right) \tag{37}$$

$$= \sum_{(\mathbf{x^m}, \mathbf{y}^m) \in \mathcal{D}} D_{\mathrm{KL}}(\delta(\mathbf{y}|\mathbf{y}^m) \| p(\mathbf{y}|\mathbf{x}^m; w)) \tag{38}$$

where $\delta(\mathbf{y}|\mathbf{y}^m) = 1$ at $\mathbf{y} = \mathbf{y}^m$ and 0 otherwise. The optimal w is achieved when $\delta(\mathbf{y}|\mathbf{y}^m) = p(\mathbf{y}|\mathbf{x}^m; w)$, which the ground-truth has 1.0 probability, and all negative outputs have 0 probability.

3.1 Expected Reward Maximization with Entropy Regularization

It is intuitive that we may find a good parameter w such that $y \in Y$ which are close to ground-truth y^* has higher conditional probability than those far different from the ground-truth, and such w could better capture the energy landscape. But, how can we achieve it ? Motivated by the (batch-mode) reinforcement learning on structured prediction, whose objective is to maximize the expected reward or minimize the negative expected reward:

$$\mathcal{L}_{\mathrm{RL}}(\boldsymbol{w}; \tau, \mathcal{D}) = \sum_{(\mathbf{x}^m, \mathbf{y}^m) \in \mathcal{D}} \left\{ -\sum_{\mathbf{y} \in \mathcal{Y}} p_\theta(\mathbf{y}|\mathbf{x}^m) \cdot r\left(\mathbf{y}, \mathbf{y}^m\right) \right\} \tag{39}$$

where $r(y, y^m) \geq 0$ denotes the reward. Sometimes it is preferred to add an entropy regularizer:

$$\mathcal{L}_{\mathrm{RL}}(\boldsymbol{w}; \tau, \mathcal{D}) = \sum_{(\mathbf{x}^m, \mathbf{y}^m) \in \mathcal{D}} \left\{ -\tau \cdot \mathbb{H}\left(p(\mathbf{y}|\mathbf{x}^m; w)\right) - \sum_{\mathbf{y} \in \mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m; w) \cdot r\left(\mathbf{y}, \mathbf{y}^m\right) \right\} \tag{40}$$

where τ controls the degree of regularization, and $\mathbb{H}((p(\mathbf{y}|\mathbf{x}^m; w)))$ is the entropy:

$$\mathbb{H}\left(p(\mathbf{y}|\mathbf{x}^m; w)\right) = -\sum_{\mathbf{y} \in \mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m; w) \cdot \log p(\mathbf{y}|\mathbf{x}^m; w) \tag{41}$$

3.2 Relation to Reward Augmented Maximum Likelihood

We will show the connection between the expected reward maximization with entropy regularization and the reward-augmented conditional log-likelihood maximization. In the previous section, we discuss the risk-augmented conditional likelihood minimization without the log, but we can also add the log since log is monotone increasing. Then in risk minimization, we want to adjust the w such that large risk corresponds to small log-likelihood, and we want to minimize the risk-augmented likelihood. From the reward maximization perspective, we want to adjust the w such that large reward corresponds to small negative log-likelihood (large log-likelihood), and thus we want to minimize the reward-augmented negative likelihood.

We can further reformulate the objective as below:

$$\mathcal{L}_{\mathrm{RL}}(\boldsymbol{w};\tau,\mathcal{D}) = \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ -\tau\cdot\mathbb{H}\left(p(\mathbf{y}|\mathbf{x}^m;w)\right) - \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot r\left(\mathbf{y},\mathbf{y}^m\right) \right\} \tag{42}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\log p(\mathbf{y}|\mathbf{x}^m;w) - \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau} \right\} \tag{43}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log p(\mathbf{y}|\mathbf{x}^m;w) - \frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau}\right] \right\} \tag{44}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log p(\mathbf{y}|\mathbf{x}^m;w) - \log\exp\left(\frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau}\right)\right] \right\} \tag{45}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log p(\mathbf{y}|\mathbf{x}^m;w) - \log\exp\left(\frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau}\right) - \log\frac{1}{Z(y^m;\tau)} + \log\frac{1}{Z(y^m;\tau)}\right] \right\} \tag{46}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log p(\mathbf{y}|\mathbf{x}^m;w) - \log\underbrace{\left(\frac{1}{Z(\mathbf{y}^m;\tau)}\exp\left(\frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau}\right)\right)}_{q(\mathbf{y}|\mathbf{y^m};\tau)} + \underbrace{\log\frac{1}{Z(y^m;\tau)}}_{const.}\right] \right\} \tag{47}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log p(\mathbf{y}|\mathbf{x}^m;w) - \log q(\mathbf{y}|\mathbf{y^m};\tau)\right] \right\} + const. \tag{48}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ \sum_{\mathbf{y}\in\mathcal{Y}} p(\mathbf{y}|\mathbf{x}^m;w)\cdot\left[\log\frac{p(\mathbf{y}|\mathbf{x}^m;w)}{q(\mathbf{y}|\mathbf{y^m};\tau)}\right] \right\} + const. \tag{49}$$

$$= \tau \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} D_{\mathrm{KL}}\left(p(\mathbf{y}|\mathbf{x}^m;w)\|q(\mathbf{y}|\mathbf{y^m};\tau)\right) + const. \tag{50}$$

$$\tag{51}$$

where $Z(\mathbf{y}^m;\tau) = \sum_{\mathbf{y}\in\mathcal{Y}}\exp(r\left(\mathbf{y},\mathbf{y}^m\right)/\tau)$, and $q(\mathbf{y}|\mathbf{y^m};\tau)$ is defined as the **exponentiated payoff distribution**:

$$q(\mathbf{y}|\mathbf{y^m};\tau) = \frac{1}{Z(\mathbf{y}^m;\tau)}\exp\left(\frac{r\left(\mathbf{y},\mathbf{y}^m\right)}{\tau}\right) \tag{52}$$

We can see that minimizing the $\mathcal{L}_{\mathrm{RL}}$ is same as minimizing the D_{KL}. However, minimizing the $\mathcal{L}_{\mathrm{RL}}$ is challenging because of the large variance of the gradients, and minimizing the D_{KL} by using $q(\mathbf{y}|\mathbf{y^m};\tau)$ to approximate $p(\mathbf{y}|\mathbf{x}^m;w)$ is also difficult since w is not fixed. Maybe we can exploit the f-GAN to minimize the KL-Divergence here.

Alternatively, we may try to optimize the KL-divergence in opposite directions, that is

$$D_{\mathrm{KL}}\left(q(\mathbf{y}|\mathbf{y^m};\tau)\|p(\mathbf{y}|\mathbf{x}^m;w)\right), \tag{53}$$

since they both have the same global optimum of p_w. In addition, we can derived that

$$D_{\mathrm{KL}}\left(q(\mathbf{y}|\mathbf{y^m};\tau)\|p(\mathbf{y}|\mathbf{x}^m;w)\right) = \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ -\sum_{\mathbf{y}\in\mathcal{Y}} q\left(\mathbf{y}|\mathbf{y}^m;\tau\right)\log p_\theta(\mathbf{y}|\mathbf{x}) \right\} - \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \mathbb{H}\left(q\left(\mathbf{y}|\mathbf{y}^m,\tau\right)\right) \tag{54}$$

where we can view the first term as **Reward Augmented Maximum Likelihood** objective and second term as constant value:

$$\mathcal{L}_{\mathrm{RAML}}(w;\tau,\mathcal{D}) = \sum_{(\mathbf{x}^m,\mathbf{y}^m)\in\mathcal{D}} \left\{ -\sum_{\mathbf{y}\in\mathcal{Y}} q\left(\mathbf{y}|\mathbf{y}^m;\tau\right)\log p_\theta(\mathbf{y}|\mathbf{x}) \right\} \tag{55}$$

which is our desirable objective at the very beginning. Therefore, we build the connection between the reinforcement learning objective with entropy regularization on structured prediction problems and the reward augmented maximum likelihood, where the latter is more easier to train.

In summary, the key problem here is to do the distribution approximation, and two direction of KL-Divergence lead to two different objectives.

3.3 Optimization and Sampling

To optimize the reward augmented maximum likelihood objective, $\mathcal{L}_{\text{RAML}}(w;\tau)$, we need to

1. draw unbiased samples from $q(y|y^*;\tau)$; given mini-batch y^*, we draw some y samples.
2. Based on those samples, we can estimate the $\nabla_w \mathcal{L}_{\text{RAML}}(w;\tau)$

$$\nabla_w \mathcal{L}_{\text{RAML}}(w;\tau) = \mathbb{E}_{q(\mathbf{y}|\mathbf{y}^m;\tau)}\left[-\nabla_w \log p(\mathbf{y}|\mathbf{x}^m;w)\right] \tag{56}$$

   But how many samples are enough ? How can we draw samples from $q(y|y^*;\tau)$; similarly, how can we draw samples from an exponential distribution [2]?
3. If we can draw N samples from $q(y|y^m;\tau)$ for each y^m, then we can compute the

$$\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{n=1}^{N} -\nabla_w \log p(y^m|x^m;w) \tag{57}$$

   and apply stochastic gradient descent.

For each ground-truth output y^*, we need to sample auxiliary outputs:

1. We sample from $q(y|y^*;\tau)$ by stratified sampling, where we first select a particular distance, and then sample an output with that distance value.
2.

References

[1] Cvpr 2011 tutorial: Structured prediction and learning in computer vision. 2011. http://www.nowozin.net/sebastian/cvpr2011tutorial/.

[2] Introduction to sampling methods. 2012. http://www.cse.psu.edu/~rtc12/CSE586/lectures/cse586samplingPreMCMC.pdf.

[3] Nan Ding and Radu Soricut. Cold-start reinforcement learning with softmax policy gradient. In *Advances in Neural Information Processing Systems*, pages 2817–2826, 2017.

[4] Kevin Gimpel and Noah A Smith. Softmax-margin crfs: Training log-linear models with cost functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 733–736. Association for Computational Linguistics, 2010.

[5] Mohammad Norouzi, Samy Bengio, Navdeep Jaitly, Mike Schuster, Yonghui Wu, Dale Schuurmans, et al. Reward augmented maximum likelihood for neural structured prediction. In *Advances In Neural Information Processing Systems*, pages 1723–1731, 2016.

[6] Bowen Tan, Zhiting Hu, Zichao Yang, Ruslan Salakhutdinov, and Eric Xing. Connecting the dots between mle and rl for sequence generation. *arXiv preprint arXiv:1811.09740*, 2018.