Homework 1
600.482/682 Deep Learning
Spring 2019

Kemeng Zhang

February 17, 2019

Due Mon 2/18 11:59pm.
Please type your answers inline of the LaTeX file
Submit PDF to Gradescope with entry code MYRR74

1. A doctor and a resident are reading scans and classifying tumors. Given 10 scans, the doctor classifies 9 of them correctly, while the resident classifies 6 correctly.

   (a) Give the formula for probability and odds. Explain their difference.
   $P(A) = \frac{\#ofEventA}{Total\#ofEvents}$
   $Odd(A) = \frac{\#ofEventA}{Total\#ofNot\ A\ Events}$
   Probability is between 0 and 1, but odd is a ratio between 0 to ∞

   (b) What is the odds of the doctor reading the scan correctly? What is the odds of the resident? What is the odds ratio of the doctor reading the scan correctly compared to the resident?
   $Odd(Doctor\ reading\ the\ scan\ correctly) = \frac{9}{10-9} = 9$
   $Odd(Resident\ reading\ the\ scan\ correctly) = \frac{6}{10-6} = 1.5$
   odds ratio $= \frac{9}{1.5} = 6$

   (c) What is a logit and how can it be used to derive a linear model to express the exponent of odds?
   $\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = \log(p) - \log(1-p) = Wx + b$

   (d) Using the model, what is the odds ratio of making a correct reading as a doctor compared to a resident?
   $\text{Odds} = \frac{P}{1-P}$
   $\text{OddsRatio} = \frac{\text{Odds}(Doctor)}{\text{Odds}(Resident)} = \exp\left(\log\left(\frac{p_1/(1-p_1)}{p_2/(1-p_2)}\right)\right) =$
   $\exp\left(\log(p_1) - \log(1-p_1) - \left(\log(p_2) - \log(1-p_2)\right)\right) = 6$
   $P_1 = 0.9$
   $P_2 = 0.6$

   (e) Given the previous odds ratio, derive the probability expressing how much more likely the doctor is to make the correct classification compared to the resident.
   5

2. Problems for the maximum likelihood estimate and the maximum a posteriori estimate:

   (a) Assume $p(y \mid x) = N(ax, s^2)$, where all quantities are scalars and where a and s are known constants. You observe $y_1, \ldots, y_N$. Derive the maximum likelihood estimate (MLE) of x.
   Let $\mu = ax$
   $L(\mu, s^2) = \prod_{i=1}^{N} f(y_i; \mu, s^2) = \left(\sqrt{2\pi s^2}\right)^{-N} \exp\left[-\frac{1}{2s^2}\sum_{i=1}^{N}(y_i - \mu)^2\right]$

$\log L(\mu, s^2) = -\frac{N}{2}\log s^2 - \frac{N}{2}\log(2\pi) - \frac{\sum(y_i-\mu)^2}{2s^2}$
$0 = \frac{\partial}{\partial \mu}\log\Big(L(\mu, s^2)\Big) = \frac{2\sum(y_i-\mu)}{2s^2}.$
$\mu_{MLE} = \bar{y}$
$x_{MLE} = \bar{y}/a$

(b) Assume $p(y \mid x) = N(ax, s^2)$, where all quantities are scalars, a and s are known constants, and the prior distribution over x is $N(m, r^2)$, where m and r are known constants. The *maximum a posteriori (MAP)* estimate is the value of x that maximizes $P(x|y) = P(y|x)P(x)/P(y)$. You observe $y_1, \ldots, y_N$. Derive the maximum a posteriori estimate of x.
Let $\mu = ax$
$\underset{x}{\text{argmax}} P(x|y) = \underset{x}{\text{argmax}} P(y|x)P(x)/P(y) = \underset{x}{\text{argmax}} P(y|x)P(x) = \underset{x}{\text{argmax}} \log P(y|x)P(x)$
$\log P(y|x)P(x) = -\frac{N}{2}\log s^2 - \frac{N}{2}\log(2\pi) - \frac{\sum(y_i-\mu)^2}{2s^2} - \frac{1}{2}\log r^2 - \frac{1}{2}\log(2\pi) - \frac{(\mu-m)^2}{2r^2}$
$0 = \frac{\partial}{\partial \mu}\log\Big(\log P(y|x)P(x)\Big) = \frac{2\sum(y_i-\mu)}{2s^2} - \frac{\mu-m}{r^2}.$
Solve for $\mu = \frac{N\bar{y}r^2+ms^2}{Nr^2+s^2}$
$x = \frac{N\bar{y}r^2+ms^2}{Nr^2+s^2}/a$

(c) Assume that $a = 2$, $s = 3$, $m = 1$, and $r = 0.5$, and that you observed y values of -0.85, 0.68, -1.26, 2.36, 1.27, -3.49, -0.54, and 0.12. What are the maximum likelihood and maximum a posteriori estimates of x? Explain the difference you observe between x_{MLE} and x_{MAP}?
$x_{MLE} = \bar{y}/a = -0.107$
$x_{MAP} = \frac{N\bar{y}r^2+ms^2}{Nr^2+s^2}/a = 0.390$
MAP Estimation is a bayesian method, where you have prior knowledge of your parameter, called the "prior". You then update your prior belief using incoming data, to get a posterior distribution of the parameter. Posterior distribution can differ based on the prior. MLE only selects the most likely parameter based on likelihood.

3. Recall in class, we learned the form of a linear classifier as $f(\boldsymbol{x}; \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$. In order to learn the parameters, we need to perform error-backpropagation, a way to compute partial derivatives (or gradients) w.r.t. the parameters of a neural network. Here, we are interested in the derivative of the softmax loss for a multinomial classification problem.
Let's first define the notations:

input features :	$\boldsymbol{x} \in \mathbb{R}^D$.
target labels :	$\boldsymbol{y} \in \mathbb{R}^K$.
multinomial linear classifier :	$\boldsymbol{f} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}, \quad \boldsymbol{W} \in \mathbb{R}^{K\times D}$ and $\boldsymbol{f}, \boldsymbol{b} \in \mathbb{R}^K$
e.g., for the k-th classification :	$f_k = \boldsymbol{w}_k^T\boldsymbol{x} + b_k$, corresponding to y_k, where $\boldsymbol{w}_k^T$ is the k-th row of $\boldsymbol{W}, k \in \{1...K\}$

(a) Please express the softmax loss of logistic regression, $L(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{b}, y)$ using the above notations.
$L = -\log\big(h(x)\big) = -\log\big(\frac{e^{f_i}}{\sum_{j=1}^K e^{f_j}}\big) = -\log\big(\frac{e^{\boldsymbol{w}_i^T\boldsymbol{x}+b_i}}{\sum_{j=1}^K e^{\boldsymbol{w}_j^T\boldsymbol{x}+b_j}}\big)$

(b) Please calculate its derivative Jacobian $\frac{\partial L}{\partial \boldsymbol{w}_k}$.

$$\frac{\partial L}{\partial \boldsymbol{w}_k} = -\frac{\partial}{\partial \boldsymbol{w}_k}\log\big(\frac{e^{\boldsymbol{w}_i^T\boldsymbol{x}+b_i}}{\sum_{j=1}^K e^{\boldsymbol{w}_j^T\boldsymbol{x}+b_j}}\big)$$
$$= -\frac{\partial}{\partial \boldsymbol{w}_k}\big(\boldsymbol{w}_i^T\boldsymbol{x} + b_i\big) + \frac{\partial}{\partial \boldsymbol{w}_k}\log\big(\sum_{j=1}^k e^{\boldsymbol{w}_j^T\boldsymbol{x}+b_j}\big)$$

When k = i:

$$\frac{\partial L}{\partial \boldsymbol{w}_k} = -\boldsymbol{x} + \frac{e^{\boldsymbol{w}_k^T \boldsymbol{x} + b_i}}{\sum_{j=1}^{K} e^{\boldsymbol{w}_j^T \boldsymbol{x} + b_j}} \boldsymbol{x}$$

When $k \neq i$:

$$\frac{\partial L}{\partial \boldsymbol{w}_k} = \frac{e^{\boldsymbol{w}_k^T \boldsymbol{x} + b_i}}{\sum_{j=1}^{K} e^{\boldsymbol{w}_j^T \boldsymbol{x} + b_j}} \boldsymbol{x}$$

4. Direction of KL divergence. In many real-world applications, we often don't have full observation of the target distribution. Then it is important to determine the direction of KL divergence when choosing it as an objective function. Here, we want to show the difference of KL divergence directions by calculating the gradient.

   (a) Show that KL divergence is asymmetric using the following example. We define a discrete random variable X. Now consider the case that we have two sampling distribution $P(x)$ and $Q(x)$, which we present as two hard encoded vector:

   $$P(x) = [1,\ 6,\ 12,\ 5,\ 2,\ 8,\ 12,\ 4]$$
   $$Q(x) = [1,\ 3,\ 6,\ 8,\ 15,\ 10,\ 5,\ 2]$$

   Please compute 1) discrete probability distribution, $p(x)$ and $q(x)$. (hint: calculate the normalization). 2) two directions of KL divergence, $\mathbf{KL}(p||q)$ and $\mathbf{KL}(q||p)$.
   $p(x) = [0.02, 0.12, 0.24, 0.1, 0.04, 0.16, 0.24, 0.08]$
   $q(x) = [0.02, 0.06, 0.12, 0.16, 0.3, 0.2, 0.1, 0.04]$
   $\mathbf{KL}(p||q) = -\sum_{x \in \mathcal{X}} p(x) \log\left(\frac{q(x)}{p(x)}\right) = 0.35$
   $\mathbf{KL}(q||p) = 0.48$

   (b) Next, we try to optimize a model to fit the target distribution. We hope to pay attention to the issue of normalization and see what elements are involved in each direction.
   Note that p and q_θ are probability distributions. To simplify expression, $p(d)$ and $q(d)$ are all discrete variables, where $p(d) = P(d)/Z_p$, Z_p is normalization factor. $p(d)$ is regarded as the target distribution, and we optimize θ to fit model distribution $q_\theta(d)$ to $p(d)$. Please express $\mathbf{KL}(q_\theta||p)$ and $\mathbf{KL}(p||q_\theta)$ as optimization objective functions. (hint: remove all constant items that are not related to the optimization process)
   $\underset{\theta}{\operatorname{argmin}}\, \mathbf{KL}(q_\theta||p)$ or
   $\underset{\theta}{\operatorname{argmin}}\, \mathbf{KL}(p||q_\theta)$

   (c) Can you tell which direction is easier for computation? Why? Then please calculate the gradient of $\mathbf{KL}(q_\theta||p)$ and $\mathbf{KL}(p||q_\theta)$ w.r.t. $q_\theta(d)$ using the results in (b).
   $\mathbf{KL}(p||q_\theta)$ is easier for computation. $q(x)$ only appears once and is the numerator in log.
   $\frac{\partial}{\partial q_\theta(d)} \mathbf{KL}(p||q_\theta) = -\sum_{x \in \mathcal{X}} \frac{p(x)}{q_\theta(x)}$

5. In this problem, you are provided an opportunity to perform hands-on calculation of the SVM loss and softmax loss we learned in class.
   We define a model of Linear Classifier:

   $$f(\boldsymbol{x}, \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$$

   Giving a data sample:

   $$\boldsymbol{x}_i = \begin{bmatrix} -15 \\ 22 \\ -44 \\ 56 \end{bmatrix},\ y_i = 2$$

   At one iteration, we have

   $$\boldsymbol{W} = \begin{bmatrix} 0.01,\ -0.05,\ 0.1,\ 0.05 \\ 0.7,\ 0.2,\ 0.05,\ 0.16 \\ 0.0,\ -0.45,\ -0.2,\ 0.03 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 0.0 \\ 0.2 \\ -0.3 \end{bmatrix}$$

Please calculate 1) SVM Loss (hinge loss) and 2) softmax loss (cross-entropy loss) of this sample point.

$$f(\boldsymbol{x_i}, \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x_i} + \boldsymbol{b} = \begin{bmatrix} -2.85 \\ 0.86 \\ 0.28 \end{bmatrix}$$

$$L_i = \max(0, -2.85 - 0.86 + 1) + \max(0, 0.28 - 0.86 + 1) = 0.42$$

$$\text{softmax}(\boldsymbol{x_i}) = -\log \frac{e^{0.86}}{e^{0.86} + e^{-2.85} + e^{0.28}} = 0.46$$