MSBD 5012 Machine Learning **(Due: 14th October)**

Written Assignment #2 Solution

Instructor: Nevin L. Zhang *Name:* studentname, *ID:* studentID

Problem 1: Naive Bayes

Answer:

(a)

$$P(y|x_1,x_2) = \frac{P(x_1,x_2|y)P(y)}{P(x_1,x_2)} = \frac{P(x_1|y)P(x_2|y)P(y)}{P(x_1,x_2)}$$

$$P(y=0) = \frac{3}{8}, \quad P(y=1) = \frac{5}{8}$$

$$P(x_1=1|y=0) = 1, \quad P(x_1=0|y=0) = 0$$

$$P(x_2=1|y=0) = \frac{1}{3}, \quad P(x_2=0|y=0) = \frac{2}{3}$$

$$P(x_1=1|y=1) = \frac{1}{5}, \quad P(x_1=0|y=1) = \frac{4}{5}$$

$$P(x_2=1|y=1) = \frac{3}{5}, \quad P(x_2=0|y=1) = \frac{2}{5}$$

(b)

Instance 1:

$$P(y=1|x_1=0,x_2=0) = \frac{P(x_1=0|y=1)P(x_2=0|y=1)P(y=1)}{P(x_1=0,x_2=0)}$$

$$= \frac{P(x_1=0|y=1)P(x_2=0|y=1)P(y=1)}{P(x_1=0,x_2=0|y=0)P(y=0) + P(x_1=0,x_2=0|y=1)P(y=1)}$$

$$= \frac{P(x_1=0|y=1)P(x_2=0|y=1)P(y=1)}{P(x_1=0|y=0)P(x_2=0|y=0)P(y=0) + P(x_1=0|y=1)P(x_2=0|y=1)P(y=1)} = 1$$

$$P(y=0|x_1=0,x_2=0) = 1 - P(y=1|x_1=0,x_2=0) = 0$$

Instance 7:

$$P(y=1|x_1=1,x_2=1) = \frac{P(x_1=1|y=1)P(x_2=1|y=1)P(y=1)}{P(x_1=1,x_2=1)}$$

$$= \frac{P(x_1=1|y=1)P(x_2=1|y=1)P(y=1)}{P(x_1=1,x_2=1|y=0)P(y=0) + P(x_1=1,x_2=1|y=1)P(y=1)}$$

$$= \frac{P(x_1=1|y=1)P(x_2=1|y=1)P(y=1)}{P(x_1=1|y=0)P(x_2=1|y=0)P(y=0) + P(x_1=1|y=1)P(x_2=1|y=1)P(y=1)} = \frac{3}{8}$$

$$P(y=0|x_1=1,x_2=1) = 1 - P(y=1|x_1=1,x_2=1) = \frac{5}{8}$$

Problem 3: FNN, Forward and Backward Propagation

Answer:

(a)

$$h_{11} = w_{11}^{(1)}x_1 + w_{21}^{(1)}x_2 = -1, \quad u_{11} = \tanh(h_{11}) = -0.7615$$

$$h_{12} = w_{12}^{(1)}x_1 + w_{22}^{(1)}x_2 = 1, \quad u_{12} = \tanh(h_{12}) = 0.7615$$

$$h_{21} = w_{11}^{(2)}u_{11} + w_{21}^{(2)}u_{12} = 1.523, \quad u_{21} = \tanh(h_{21}) = 0.9092$$

$$h_{22} = w_{12}^{(2)}u_{11} + w_{22}^{(2)}u_{12} = 1.523, \quad u_{22} = \tanh(h_{22}) = 0.9092$$

$$z = w_1^{(3)} u_{21} + w_2^{(3)} u_{22} = 1.8184$$

$P(y=1|x_1=1,x_2=2,\theta) = \sigma(z) = 0.8604$, $P(y=0|x_1=1,x_2=2,\theta) = 1 - P(y=0|x_1=1,x_2=2,\theta) = 0.1396$

(b)

$$\frac{\partial L}{\partial z} = \sigma(z) - y = 0.8604$$

$$\delta_{21} = \frac{\partial u_{21}}{\partial h_{21}} w_1^{(3)} \delta z = (1 - u_{21}^2) \cdot w_1^{(3)} \cdot \delta_z = 0.1492$$

$$\delta_{22} = \frac{\partial u_{22}}{\partial h_{22}} w_2^{(3)} \delta z = (1 - u_{22}^2) \cdot w_2^{(3)} \cdot \delta_z = 0.1492$$

$$\delta_{11} = \frac{\partial u_{11}}{\partial h_{11}} \left(w_{11}^{(2)} \delta_{21} + w_{12}^{(2)} \delta_{22} \right) = (1 - u_{11}^2) \left(w_{11}^{(2)} \delta_{21} + w_{12}^{(2)} \delta_{22} \right) = -0.1254$$

$$\delta_{12} = \frac{\partial u_{12}}{\partial h_{12}} \left(w_{21}^{(2)} \delta_{21} + w_{22}^{(2)} \delta_{22} \right) = (1 - u_{12}^2) \left(w_{21}^{(2)} \delta_{21} + w_{22}^{(2)} \delta_{22} \right) = 0.1254$$

$$\frac{\partial L}{\partial w_{22}^{(2)}} = \delta_{22} \cdot u_{12} = 0.1136$$

$$\frac{\partial L}{\partial w_{22}^{(1)}} = \delta_{12} \cdot x_2 = 0.2508$$

We need to reduce the two parameters.

Problem 4: Sigmoid Functions

Answer:

Sigmoid function is

$$s(x) = \frac{1}{1 + e^{-x}}$$

It has a derivative of

$$\frac{\partial s(x)}{\partial x} = s(x)(1 - s(x))$$

While $0 < s(x) < 1$, it's obvious that $0 < \frac{\partial s(x)}{\partial x} < 1$. Thus when we use the chain rule to back-propagate our parameters, we may get many items between 0 and 1. The multiplication of these items may lead to a number close to 0. This may make our gradient vanish. However, when it comes to the output, we want a probability for classification, the output of a Sigmoid function is just a number between 0 and 1, and this is just what we want.

Problem 5: Dropout

Answer:
Dropout acts as a regularization method to avoid overfitting. When adding dropout layers, some hidden neurons are randomly thrown, so the parameters are partly updated. This avoids the internal relations of the hidden neurons. It is like training different networks and make average. So it can sufficiently avoid overfitting.

Problem 6: Adam Optimization

Answer:
Adam uses the momentum mechanism to accelerate the gradient descent. Besides, same as RMSProp, Adam uses a γ to exponentially decrease the influence of past gradients. And it has a bias correction mechanism for m and v, which makes the iteration flat.